НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»
ІНСТИТУТ ПРИКЛАДНОГО СИСТЕМНОГО АНАЛІЗУ
КАФЕДРА МАТЕМАТИЧНИХ МЕТОДІВ СИСТЕМНОГО АНАЛІЗУ

На правах рукопису
УДК 004.942:519.226.3

«До захисту допущено»
В. о. завідувача кафедри ММСА
_____ О. Л. Тимощук
“___”_____ 2020 р.

Магістерська дисертація

на здобуття ступеня магістра за спеціальністю 124 Системний аналіз
на тему: «Медична діагностична система на основі байєсівських мереж»

Виконала:
студентка ІІ курсу, групи КА-91мп
Корнійчук Оксана Сергіївна _____

Керівник:
професор кафедри ММСА,
д.т.н., проф. Бідюк П. І. _____

Рецензент:
декан ФІОТ КПІ ім. Ігоря Сікорського,
професор, д.т.н. Теленик С. Ф. _____

Засвідчую, що у цій магістерській дисертації немає запозичень з праць інших авторів без відповідних посилань.
Студент _____

Київ
2020

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»
ІНСТИТУТ ПРИКЛАДНОГО СИСТЕМНОГО АНАЛІЗУ
КАФЕДРА МАТЕМАТИЧНИХ МЕТОДІВ СИСТЕМНОГО АНАЛІЗУ

Рівень вищої освіти — другий (магістерський)
Спеціальність — 124 «Системний аналіз»

ЗАТВЕРДЖУЮ
В. о. завідувача кафедри ММСА
О. Л. Тимощук
«___» _____ 2020 р.

ЗАВДАННЯ
на магістерську дисертацію студенту Корнійчук Оксані Сергіївні

1. Тема дисертації: «Медична діагностична система на основі байєсівських мереж», науковий керівник дисертації Бідюк Петро Іванович, д.т.н., професор, затверджені наказом по університету від «02» листопада 2020 року № 3182-с

2. Термін подання студентом дисертації: 16 грудня 2020 р.

3. Об'єкт дослідження: взаємозв'язки між симптомами та зовнішніми чинниками і наявністю у людини хвороби серця або COVID-19, представлені у обраних вибірках даних.

4. Предмет дослідження: байєсівські мережі для діагностики хвороб серця та COVID-19, методи їх побудови та оцінювання якості роботи.

5. Перелік завдань, які потрібно розробити:

1) дослідити актуальність проблеми підвищення якості оцінювання стану пацієнтів;

2) провести огляд існуючих методів побудови систем підтримки прийняття лікарських рішень та деяких сучасних їх реалізацій;

3) дослідити застосування байєсівських мереж у медичній діагностиці, методи їх побудови та деякі програмні середовища для реалізації;

4) пошук та обробка даних для побудови та навчання байєсівських мереж;

5) на основі оброблених даних побудувати відповідні байєсівські мережі та провести їх навчання;

6) перевірити точність роботи створених систем, навести приклади;

7) розробити стартап-проект виведення на ринок результатів дослідження;

8) розробити концептуальні висновки за результатами наукового дослідження.

6. Орієнтовний перелік графічного (ілюстративного) матеріалу:

1) інтерфейси програмних середовищ для побудови байєсівських мереж;

2) результати роботи створених систем;

3) таблиці у розділі стартап-проекту.

7. Орієнтовний перелік публікацій:

Участь у Восьмій міжнародній науково-технічній конференції з публікацією тез доповіді.

Стаття у збірнику «Системні науки і кібернетика».

8. Дата видачі завдання: 02 вересня 2020 р.

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації
1.	Концептуальний вступ дисертації. Формулювання об’єкта, предмета, цілі, завдань, новизни, практичної значущості результатів	07.09.2020—09.09.2020
2.	Перший розділ. Огляд літературно-інформаційних джерел. Понятійно-категоріальний апарат. Огляд існуючих методів побудови діагностичних систем підтримки прийняття рішень та їх сучасних реалізацій.	10.09.2020—27.09.2020
3.	Другий розділ. Теоретичні основи байєсівських мереж та обраних методів їх побудови. Огляд програмних середовищ для їх побудови.	28.09.2020—11.10.2020
4.	Третій розділ. Пошук та обробка даних. Побудова та навчання мереж Байєса. Перевірка точності їх роботи та приклади.	12.10.2020—01.11.2020
5.	Четвертий розділ. Розробка стартап-проекту.	02.11.2020—11.11.2020
6.	Концептуальні висновки. Перспективи розвитку отриманих рішень.	12.11.2020—16.11.2020
7.	Оформлення дисертації та підготовка ілюстративного матеріалу для доповіді.	17.11.2020—29.11.2020

Студент О.С. Корнійчук

Науковий керівник дисертації П.І. Бідюк

РЕФЕРАТ

Магістерська дисертація: 126 с., 21 табл., 27 рис., 4 дод., 40 джерел.

МЕРЕЖА БАЙЄСА, СИСТЕМА ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ, ДІАГНОСТИЧНА СИСТЕМА ПІДТРИМКИ ПРИЙНЯТТЯ ЛІКАРСЬКИХ РІШЕНЬ, ТЕОРЕМА БАЙЄСА, ЙМОВІРНІСНИЙ ВИСНОВОК, NPC, PC, ЖАДІБНИЙ АЛГОРИТМ ПОШУКУ ТА ОЦІНКИ, ДЕРЕВО ЧУ-ЛІУ, ПОЛІДЕРЕВО РІБАНА-ПЕРЛА, TREE AUGMENTED NAIVE BAYES.

Об’єктом дослідження є взаємозв’язки між симптомами та зовнішніми чинниками і наявністю у людини хвороби серця або COVID-19, представлені у обраних наборах даних.

Предметом дослідження є байєсівські мережі для діагностики хвороб серця та COVID-19, методи їх побудови та оцінювання якості роботи.

Метою роботи є створення систем підтримки прийняття рішень при діагностиці вказаних хвороб на основі байєсівських мереж та перевірка ефективності їх роботи.

Методи дослідження: NPC, PC, жадібний алгоритм пошуку та оцінки, дерево Чу-Ліу, полідерево Рібана-Перла, Tree Augmented Naive Bayes, алгоритм Hugin формування ймовірнісного висновку.

У роботі проведено огляд існуючих методів побудови діагностичних систем підтримки прийняття рішень та деяких сучасних їх реалізацій. Також проаналізовано роль байєсівських мереж у медичній діагностиці, розглянуто деякі методи їх побудови та програмні середовища для реалізації.

Науковою новизною є нові моделі у формі БМ, дві системи підтримки прийняття рішень при діагностиці наявності хвороб серця та COVID-19 відповідно, створені на основі байєсівських мереж, навчених за обраними вибірками даних. Вони можуть бути використані для попередньої діагностики вказаних захворювань медичними працівниками.

ABSTRACT

Master's thesis: 126 p., 21 tabl., 27 fig., 4 appendixes, 40 sources.

BAYESIAN NETWORK, DECISION SUPPORT SYSTEM, DIAGNOSIS DECISION SUPPORT SYSTEM, BAYES' THEOREM, INFERENCE, NPC, PC, GREEDY SEARCH-AND-SCORE ALGORITHM, CHOW-LIU TREE, REBANE-PEARL POLYTREE, TREE AUGMENTED NAIVE BAYES.

The object of the study are the relationships between symptoms and external factors and the presence of human heart disease or COVID-19, presented in selected datasets.

The subject of the study are Bayesian networks for the diagnosis of heart disease and COVID-19, methods for their construction and evaluation of quality of work.

The aim of the work is to create decision support system for the diagnosis of these diseases on the basis of Bayesian networks and to verify the effectiveness of their work.

Research methods: NPC, PC, greedy search-and-score algorithm, Chow-Liu tree, Riban-Pearl polytree, Tree Augmented Naive Bayes, Hugin algorithm for forming a probabilistic conclusion.

The paper reviews the existing methods of building diagnostic decision support systems and some of their modern implementations. The role of Bayesian networks in medical diagnostics is also analyzed, some methods of their construction and software environments for implementation are considered.

A scientific novelty is two decision support systems for diagnosing the presence of heart disease and COVID-19, respectively, created on the basis of Bayesian networks trained on selected data sets. They can be used for preliminary diagnosis of these diseases by the health professionals.

ЗМІСТ

ПЕРЕЛІК ПРИЙНЯТИХ СКОРОЧЕНЬ

БВШ – багатовимірне шкалювання

БІК – байєсівський інформаційний критерій

БМ – байєсівська мережа

МГК – метод головних компонент

ОМД – опис мережі мінімальної довжини

ОПР – особа, що приймає рішення

ПЗ – програмне забезпечення

САГ – спрямований ациклічний граф

СППЛР – система підтримки прийняття лікарських рішень

СППР – система підтримки прийняття рішень

AIC (англ. Akaike Information Criterion) – інформаційний критерій Акайке

BIC (англ. Bayesian Information Criterion) – байєсівський інформаційний критерій

COVID-19 (англ. CoronaVirus Disease 2019) – коронавірусна хвороба 2019

DSS (англ. Decision Support System) – система підтримки прийняття рішення

EHR (англ. Electronic Health Record) – електронні медичні картки

NPC (англ. Necessary Path Condition) –умова необхідного шляху

PC (англ. Path Condition) – умова для шляху

TAN (англ. Tree Augmented Naive Bayes) – аївний байєсівський класифікатор з додаванням дерева

ВСТУП

Незважаючи на світовий технічний прогрес та значні досягнення у всіх сферах людського життя, зокрема й у медицині, деякі проблеми й досі не втрачають своєї актуальності. Серед них є й підвищення якості оцінювання стану пацієнтів. Для постановки правильного та своєчасного діагнозу лікарю потрібно за короткий час врахувати дуже багато різних факторів, таких як явні та приховані симптоми, результати аналізів, попередні захворювання тощо. До того ж у таких ситуаціях завжди присутня невизначеність та неповнота наявної інформації. Зважаючи на всі ці фактори, можна зробити висновок, що задача оцінювання стану пацієнта є дуже складною. Саме тому з розвитком сучасних технологій на допомогу лікарям прийшли системи підтримки прийняття лікарських рішень. Завдяки базам даних та вбудованим алгоритмам, вони можуть швидко проаналізувати великий об’єм наукової літератури або історичні дані щодо наявних прецедентів та надати лікарю рекомендації щодо діагнозу та можливого лікування. Такі системи будуються на основі різних алгоритмів, зокрема на штучних нейронних мережах, системах логічних правил, а також на основі байєсівських мереж. Головною перевагою мереж Байєса є те, що вони коректно враховують причинність досліджуваного процесу. Це є однією із причин їхнього широкого застосування у медичній діагностиці, а саме у системах підтримки прийняття лікарських рішень.

У першому розділі роботи аналізується актуальність проблеми підвищення якості оцінювання стану пацієнта, проводиться огляд результатів відомих досліджень та методів побудови діагностичних СППЛР. Також наведені приклади деяких сучасних систем для медичної діагностики.

У другому розділі наводяться теоретичні основи байєсівських мереж, зокрема їх методика побудови. Також розглядаються алгоритми побудови та оцінювання структури БМ і метод Hugin формування ймовірнісного

висновку. Проводиться огляд застосування байєсівських мереж у медичній діагностиці, наводяться деякі програмні продукти для їх побудови.

У третьому розділі аналізуються обрані набори даних на наявність пропусків, зв’язки між змінними тощо. На основі оброблених даних створюються та навчаються байєсівські мережі, перевіряється точність їхньої роботи, наводяться приклади.

У четвертому розділі проаналізовано доцільність запуску створених діагностичних систем як стартап-проекту.

РОЗДІЛ 1
АНАЛІЗ ПРОБЛЕМИ ПІДВИЩЕННЯ ЯКОСТІ ОЦІНЮВАННЯ СТАНУ ПАЦІЄНТА

На сьогодні, не зважаючи на розвиток технологій діагностики якість оцінювання стану пацієнта й досі потребує підвищення. Одним із засобів, що цьому сприяють, є системи підтримки прийняття лікарських рішень.

У цьому розділі проводиться аналіз актуальності зазначеної проблеми, здійснюється огляд результатів відомих досліджень та методів побудови СППЛР. Також розглянуто деякі сучасні системи, які використовуються для медичної діагностики та оцінювання стану пацієнта.

1.1 Аналіз актуальності проблеми підвищення якості оцінювання стану пацієнтів

На жаль, життя людини нерозривно пов’язане з хворобами, подолання яких потребує своєчасного реагування. Будь-яке лікування починається із постановки діагнозу, тому він повинен бути точним, правильним та якомогашвидше визначеним. Багато хвороб мають спільні симптоми, деякі захворювання бувають дуже рідкісними, а інші потребують миттєвого лікування. Для постановки діагнозу лікар повинен врахувати дуже багато різних факторів, саме тому системи підтримки прийняття рішень у цій сфері є дуже актуальними.

На наявність хвороби можуть вказувати багато симптомів. Проте деякі з них можуть бути тимчасовими і виникати навіть у здорових людей. Тому

система, що буде допомагати лікарю діагностувати наявність або відсутність хвороби у пацієнта за заданими симптомами, повинна бути точною, швидкою та зрозумілою.

Лікарською помилкою зазвичай називають помилку медичного працівника у постановці діагнозу або лікуванні пацієнта, що трапилася з якої-небудь причини, що виключає злий намір, халатність або недбалість лікаря.

Види лікарських помилок:

— лікувально-технічні (недостатнє для постановки правильного діагнозу обстеження пацієнта);
— діагностичні (неправильний діагноз);
— організаційні (погано організовані робоче місце і процес лікування);
— лікувально-тактичні (невірний вибір засобів і методів лікування захворювання);
— помилки в поведінці;
— неправильне ведення документації.

Діагностичні помилки знаходяться в трійці найпоширеніших.

Наслідком недбалості з боку медпрацівника може стати заподіяння шкоди пацієнту, яка Всесвітньою організацією охорони здоров’я визначається як „інцидент, що спричиняє шкоду пацієнту, таку як порушення структури або функцій організму та / або будь-який шкідливий ефект, що виникає внаслідок або пов’язаний із планами чи діями, вжитими під час надання медичної допомоги, а не є основним захворюванням або травмою, і може бути фізичним, соціальним чи психологічним (наприклад, хвороба, травма, страждання, інвалідність та смерть)" [1].

У дослідженні [2], яке включає в себе дані 337 025 пацієнтів, зазначається, що у близько 12% випадків було виявлено медичні помилки, наслідком яких стали інвалідність або смерть.

За даними дослідження Інституту Джона Хопкінса, щороку більше 250000 людей у Сполучених Штатах Америки помирають внаслідок

лікарських помилок. Таким чином вони займають третє місце серед причин смерті після хвороб серця та раку.

У країнах Європи показники теж невтішні. Щороку внаслідок неправильно поставленого діагнозу або лікування помирають 70 000 британців, 50 000 італійців, 25 000 німців, 8 000 угорців, 7 000 болгар, 3 000 іспанців [3].

В Україні, за даними МОЗ (міністерства охорони здоров'я), у результаті лікарських помилок щодня помирає 5-7 пацієнтів, а 25-30 стають інвалідами [4].

Таким чином можемо сказати, що смертність від помилок медпрацівників є дуже недооціненою. Саме тому системи, що будуть допомагати лікарям оцінювати стан пацієнтів є дуже необхідними.

1.2 Огляд результатів відомих досліджень

Загалом під терміном «система підтримки прийняття рішень» (Decision Support System, DSS) мається на увазі комп'ютерна система, яка шляхом збору та аналізу інформації може впливати на процеси прийняття рішень в різних областях людської діяльності [5, 6]. У медичній галузі такі системи називаються «системами підтримки прийняття лікарських рішень» (СППЛР).

Робоче визначення було запропоновано Робертом Хейвордом (Robert Hayward), співробітником Центру доказової медицини (Centre for Health Evidence): «Системи підтримки прийняття лікарських рішень пов'язують результати клінічних досліджень з даними конкретного пацієнта, впливаючи на вибір лікарського рішення для більш ефективного надання медичної допомоги» [7].

В результаті аналізу публікацій у науковій літературі за темою СППЛР було виявлено, що розробки і дослідження в цій області ведуться в багатьох країнах і в різних напрямках [6, 8, 9, 10] більше 40 років.

В необхідності інтелектуальної підтримки в прийнятті лікарських рішень переконалися вже дуже давно. Першими традиційними її формами були медичні енциклопедії, довідники, монографії та інша медична література, що використовувалися ще в стародавньому світі [11].

До сучасних форм СППЛР відносять медичні бази даних, інформаційно-пошукові системи та системи обробки зображень і навіть телемедицину [12], а також мобільні додатки, включаючи різні довідники для лікарів [13].

У 1970-1990 рр. під «експертною лікарською системою» найчастіше мали на увазі реалізацію функцій, які допомагають лікареві поставити правильний клінічний діагноз. Згодом розуміння цього терміна трансформувалося і розширилося. Системами підтримки прийняття лікарських рішень стали називати програми, які допомагають клініцистам приймати найбільш ефективні рішення в процесі лікування пацієнта і, тим самим, забезпечують необхідну якість медичної допомоги, в тому числі з метою скорочення лікарських помилок, але не обмежуючись тільки цим [14].

Б. А. Кобринський в своїй роботі [15] дає таке означення СППЛР: «Системи підтримки прийняття рішень в медицині (охороні здоров'я) – це проблемно-орієнтовані системи (або програмно-апаратні комплекси), які реалізують технологію інформаційної підтримки процесів прийняття лікувально-діагностичних і / або управлінських рішень медичним персоналом».

Р. А. Раводін і М. В. Резванцев в роботі [11] подають таке визначення: «Під системою підтримки прийняття лікарських рішень (СППЛР) можна розуміти будь-яку програмну систему, що допомагає лікареві приймати обґрунтовані рішення, а не діяти тільки на основі інтуїції».

Більшість опублікованих статей на тему СППЛР, описує якісь окремі програмні або навіть апаратні рішення для певних медичних спеціальностей або, в рідкісному випадку, профілю медичної допомоги, наприклад, хірургії, дерматовенерології, педіатрії і т. ін. [16, 6, 9, 10, 11].

Проведемо короткий огляд історії розвитку систем підтримки прийняття лікарських рішень, а також розглянемо області їх застосування.

Досліджувати можливість використання штучного інтелекту в медицині почали ще в кінці 1960-х на початку 1970-х років.

Однією з перших розробок була система AAHelp (Університет Лідса, 1971), що спеціалізувалася на пошуку причин різких болів і прийнятті рішення про необхідність хірургічного втручання. В результаті проведеного аналізу було виявлено, що вона дозволяла встановити правильний діагноз в 91,8% випадків, в той час як відсоток вірних діагнозів, поставлених лікарями, склав 79,6%.

Приблизно в той же час з'явилася система INTERNIST (Пітсбургський університет), яка вирішувала питання допомоги при постановці діагнозу на основі спостережуваних симптомів.

Найвідомішою медичною експертною системою на початку 1970-х, стала система MICIN, розроблена в Стенфордському університеті, призначенням якої було надання допомоги фахівцям при постановці діагнозу і визначення лікування при інфекційних захворюваннях.

До класичних систем можемо віднести Germwatcher [17], яка використовувалася у роботі лікарями-епідеміологами; PEIRS, що використовувалася при інтерпретації звітів про хімічні патології; Puff, призначенням якої була інтерпретація результатів функціонального пульмонологічного тесту; HELP – повна госпітальна інформаційна система, що ґрунтувалася на технологіях штучного інтелекту; SETH призначена для аналізу токсичності лікарських засобів; системи в області клінічної мікробіології - Vitek2 Compact, BD Phoenix, MicroScan; ATTENDING, що

займалася пошуком помилок в пропонованому рішенні і висувала альтернативний варіант.

На сьогодні СППЛР широко застосовується в хірургії, наприклад, у серцево-судинній [18].

Ще одним прикладом призначення СППЛР є класифікація тяжкості гострого панкреатиту та прогнозування летального результату [19]. Крім того існують інші системи: VM; ABEL; AI / COAG; AI / RHEUM; ANNA; BLUE; ATTENDING; GUIDON [20, 21, 22].

Існує два основні різновиди СППЛР [23]:

- СППЛР, що базуються на знаннях, тобто засновані на наукових знаннях;
- СППЛР, що не базуються на знаннях, тобто засновані не на висновках з наукових досліджень, а, наприклад, на результатах обробки зібраних статистичних даних математичними методами.

СППЛР, що базуються на знаннях

Більшість СППЛР складаються з трьох частин - інформаційної бази, механізму логічних висновків і механізму комунікації [7]. Інформаційна база містить правила і зв'язки між даними мета аналізу, які найбільш часто приймають форму правил ЯКЩО-ТО. Якщо це система визначення впливу взаємодій лікарських препаратів, правило може мати такий вигляд: ЯКЩО призначено препарат «X» і призначено препарат «Y», ТО попередити користувача. При використанні іншого інтерфейсу досвідчений користувач може редагувати інформаційну базу для підтримки її актуальності з урахуванням появи нових лікарських препаратів. Механізм логічних висновків об'єднує правила з інформаційної бази з даними пацієнта. Механізм комунікації дозволяє системі представити результати користувачеві і забезпечує введення даних в систему [7].

СППЛР, що не базуються на знаннях

СППЛР, які не використовують наукові медичні знання, будуються та працюють на основі методів машинного навчання, що забезпечують

навчання комп'ютерних систем на підставі отриманого досвіду і / або можливість встановлення закономірностей в межах масиву клінічних даних. Зазначене усуває необхідність написання правил і експертного введення. Однак, оскільки системи, засновані на «машинному навчанні», не можуть пояснити причини генерування ними тих чи інших висновків, більшість клініцистів не використовують їх безпосередньо для постановки діагнозів з причини невпевненості в точності і достовірності результатів [7]. Проте такі системи можуть бути корисні в постдіагностичний період для розкриття певних закономірностей з метою більш глибокого аналізу.

1.3 Існуючі методи побудови діагностичних СППЛР

Як було зазначено вище, при побудові СППЛР, що базуються на знаннях, використовуються правила ЯКЩО-ТО і звичайно база даних, що містить необхідну теоретичну та практичну інформацію. Таким чином основною перевагою таких систем є те, що вони дозволяють послідовно прослідкувати процес прийняття певного рішення програмою та містять його обґрунтування. Щодо недоліків, то основними з них є збільшення похибки у постановці діагнозу при наявності пропущених або неповних даних та необхідність у створенні та підтримці необхідної бази даних.

При побудові СППЛР, які не базуються на наукових медичних знаннях, зазвичай використовують один з трьох основних видів методів або їхні комбінації: байєсівські мережі, нейронні мережі і генетичні алгоритми. Головними перевагами таких систем є автоматичне врахування нових методів та технологій лікування після введення даних про існуючі прецеденти, а також менш ресурсо- та часозатратна формалізація та підтримка бази даних [24].

Нейронні мережі, використовуючи вузли і зважені зв'язки між ними, проводять аналіз закономірностей в масиві даних пацієнтів з метою встановлення асоціацій між симптомами і діагнозами. Головною їх перевагою є здатність навчати і удосконалювати модель. Серед недоліків є необхідність у великих обсягах даних для навчання, значних обчислювальних ресурсах та можливість виявлення «хибних кореляційних залежностей».

Генетичні алгоритми працюють на основі спрощених еволюційних процесів з використанням спрямованого відбору для досягнення оптимальних результатів роботи СППЛР. Алгоритми відбору оцінюють компоненти випадкових наборів рішень проблеми. Рішення, що потрапляють наверх переліку, рекомбінуються і видозмінюються, після чого процес повторюється. Це відбувається знову і знову до тих пір, поки не виявляється потрібне рішення. З цього можемо зробити висновок, що однім із недоліків генетичних алгоритмів є проблеми швидкості збіжності та взагалі її наявності. Серед переваг простота алгоритму та ефективне розпаралелювання процесів.

Наразі СППЛР, що не базуються на знаннях, зазвичай зосереджуються на обмеженому наборі симптомів (наприклад, на симптомах одного захворювання), на відміну від таких, що використовують наукову інформацію і дозволяють діагностувати різні захворювання [7].

Виробники систем, що не базуються на знаннях, обіцяють, що вони дадуть можливість значно скоротити витрати на охорону здоров'я та послабити тиск на медичних працівників. Однак існують проблеми, що перешкоджають їх масштабному впровадженню. Це, насамперед, часозатратний і трудомісткий навчальний та обчислювальний процес, а також необхідність у великих наборах даних для підвищення точності моделей. Головною ж перешкодою є відсутність інтерпретації, оскільки системи не можуть пояснити аргументи, що лежать в основі прийнятих

рішень. Через зазначені недоліки сучасні СППЛР найчастіше базуються на знаннях [25].

1.4 Деякі сучасні системи, які можна використати для медичної діагностики та оцінювання стану пацієнта

1.4.1 Siemens Healthineers

Siemens Healthineers акцентує увагу на аналізі та інтерпретації результатів тестів.

Провідний виробник пристроїв діагностичної візуалізації, компанія Siemens Healthineers також пропонує цілий ряд програмних рішень для медичних центрів [25]. Основним продуктом підтримки прийняття рішень є система управління даними Protis з програмним забезпеченням Protis Assessment. Воно об’єднує результати тестування пацієнта з різних платформ в єдиний графічний звіт та допомагає лікарям інтерпретувати клінічні дані.

Програмне забезпечення Protis Assessment – це СППЛР, що базується на знаннях та використовує стандартні правила, встановлені експертами. Воно може здійснювати тестування ліквору, що використовується для діагностики станів, які впливають на головний та спинний мозок, оцінку функції нирок, що сприяє ранньому виявленню їх захворювань, 10-річну оцінку ризику виникнення порушень у серцево-судинній системі, оцінку вмісту заліза в крові та відповідно наявності анемії, а також оцінку поживності для визнання недостачі білка у пацієнтів. Якщо не діагностувати та не вирішити проблему, порушення харчування збільшує тривалість перебування в лікарні та пов'язані з цим витрати на лікування.

Описане програмне забезпечення надає користувачу зручні для розуміння графічні звіти, пропонує їх інтерпретації та висуває рекомендації.

Даний функціонал дозволяє значно зекономити час за рахунок зменшення кількості ручної роботи, запобігання проведенню зайвих аналізів і тестів, а також скоротити витрати на лікування.

Крім Protis, компанія пропонує наступні СППЛР [25]:

- AI RAD Companion, хмарний додаток на основі AI (Artificial intelligence, штучний інтелект), який допомагає аналізувати КТ (комп'ютерна томографія), рентген та МРТ (магнітно-резонансна томографія) сканування мозку, грудної клітки та інших органів;
- Система Prisca для розрахунку пренатального ризику на основі біохімічних маркерів, ультразвукових вимірювань та демографічних показників;
- Medicalis – хмарний механізм підтримки клінічних рішень для забезпечення відповідності програмі ВКВ (відповідності критеріям використання).

1.4.2 Philips Healthcare

Philips Healthcare відповідає за постійний моніторинг та надсилання сповіщень.

Підрозділ охорони здоров'я Philips значним чином інвестував у розробку рішень СППР для доопрацювання систем моніторингу пацієнтів компанії [25]. На даний момент п'ять інструментів доступні для використання з фірмовими моніторами IntelliVue від Philips.

Horizon Trends відстежує життєво важливі показники (температуру тіла, пульс, частоту дихання та артеріальний тиск) у режимі реального часу та попереджає про небезпечні відхилення.

ST Map фокусується на конкретній частині ЕКГ, яка називається сегментом ST. Пацієнтам із гострим коронарним синдромом із ризиком

ішемії міокарда рекомендується постійний моніторинг сегмента ST. ST Map дозволяє медичному персоналу виявляти критичні зміни та приймати швидкі рішення.

Event Surveillance відстежує до чотирьох клінічних параметрів і повідомляє лікаря, коли два, три або чотири з них виявляють критичні відхилення. Рішення дозволяє встановити певні пороги та створити «розумні сигнали тривоги» відповідно до потреб конкретного пацієнта. Наприклад, ви отримаєте сповіщення, якщо частота серцевих скорочень зміниться більш ніж на 50 відсотків протягом 60 секунд.

ProtocolWatch Sepsis перевіряє життєві показники пацієнта на відповідність критеріям сепсису та повідомляє доглядачів, коли ці критерії дотримані.

Модуль Histogram Trends представляє вимірювання пацієнта протягом тривалого періоду часу. Це дозволяє лікарю побачити, чи дає терапія чи певний препарат бажаний ефект.

1.4.3 IBM Watson Health

IBM Watson Health здійснює наскрізне управління ліками, хворобами та токсинами.

Micromedex Clinical Knowledge від IBM Watson Health - це система обґрунтування клінічних рішень, що базується на фактичних даних та використовується у понад 4500 лікарнях по всьому світу [25]. Він інтегрується з усіма широко використовуваними системами EHR (Electronic Health Record, Електронні медичні картки) та CPOE (Computerized Provider Order Entry, Комп’ютеризований запис замовлення постачальника). Модульна структура дозволяє лікарням додавати функціональність

поступово, за необхідності. В даний час Micromedex містить три великі компоненти.

Модуль управління ліками надає повну інформацію про лікарські засоби та перевіряє всі типи взаємодій між ними, визначаючи потенційно небезпечні або небажані комбінації.

Він також має калькулятор дозування ліків та надає пропозиції щодо рослинних та інших альтернативних методів лікування, які можна використовувати поряд із звичайною медициною.

Модуль управління захворюваннями та станом забезпечує швидкий доступ до інформації про лікування, мінімізуючи помилки, скорочуючи витрати на лікування та запобігаючи непотрібним обстеженням та процедурам.

Модуль управління токсикологією використовується у всіх сертифікованих американськими центрами контролю за отруєннями та відділах надзвичайних ситуацій для виявлення потенційних джерел отрут, проведення токсикологічного аналізу та прийняття швидких обґрунтованих рішень у разі розливу та впливу хімічних речовин.

У березні 2020 року IBM оголосила про партнерство з DynaMed, клінічним довідковим ресурсом, що надає швидкі відповіді на медичні питання з багатьох спеціальностей [25]. Дві компанії пропонують DynaMed and Micromedex with Watson платформу для клініцистів у пункті надання допомоги. У той же час вони продовжують продавати власні рішення окремо.

1.4.4 Cerner

Cerner займається виявленням критичних станів.

Cerner є світовим постачальником медичних інформаційних технологій [25]. Окрім наскрізних систем EHR та численних клінічних програмних

засобів, компанія постачає апаратні компоненти для зв'язку та медичні пристрої. Набір інструментів CDS (Clinical decision support, підтримка прийняття клінічних рішень) розроблений для найбільш критичних умов, коли кожна година є вирішальною.

СППЛР для виявлення сепсису спрямована на раннє розпізнавання небезпечної для життя інфекції в крові. Система була побудована після того, як співзасновник Cerner Ніл Паттерсон втратив свою невістку від сепсису, викликаного пневмонією. Вона не отримала своєчасного лікування, оскільки лікарі не змогли розпізнати ранні ознаки захворювання. На сьогоднішній день рішення щодо нагляду за сепсисом Cerner впроваджено в сотнях лікарень по всій території США. Він постійно розглядає життєві показники пацієнта, і коли виявляє характерні для сепсису закономірності, попереджає клінічну групу.

Рішення для гострої ниркової недостатності (acute kidney injury, AKI) виявляє інший небезпечний стан, який вимагає ранньої ідентифікації та оперативного лікування. Модуль шукає зміни рівня сироваткового креатиніну в сечі. Як тільки його концентрація починає небезпечно збільшуватися, система повідомляє клініцистів і робить пропозиції щодо подальших дій.

Компонент швидкого реагування інтегрується з модулями сепсису та AKI, щоб забезпечити негайне втручання, коли життєві показники починають швидко і несподівано змінюватися.

1.4.5 Change Healthcare

Change Healthcare слідкує за суворим дотриманням критеріїв на всіх рівнях медичної допомоги.

Change Healthcare, один з найбільших постачальників технологій у медичній галузі, пропонує рішення, що об’єднують лікарні, пацієнтів та платників [25]. Мережі компанії керують однією третиною всіх клінічних записів у США. Його рішення для клінічного прийняття рішень під назвою InterQual може похвалитися 40-річною історією накопичення знань. Система забезпечує доступ до доказових критеріїв – або загальновизнаних стандартів відповідної допомоги – розділених на кілька великих груп.

Критерії рівня допомоги допомагають клініцистам розпізнати симптоми, оцінити тяжкість стану пацієнта та ефективність лікування. Він рекомендує наведені нижче кроки у разі ускладнень або повільної реакції на терапію.

Критерії планування амбулаторної допомоги вказують, коли аналізи, консультації та ліки є діючими, щоб лікарні могли уникати надмірних обстежень та процедур.

Критерії поведінкового здоров’я спрямовані на потреби пацієнтів з психічними розладами та розладами, спричиненими вживанням наркотичних речовин. Модуль допомагає опікунам таких пацієнтів у виборі відповідної терапії на основі поточного функціонального стану.

Change Healthcare постійно оновлює свій вміст останніми прецендентами та новинами [25]. Він використовує систему спостереження, яка щодня автоматично аналізує понад 3000 медичних джерел та вибирає нещодавно опубліковану інформацію, яка буде перевірена експертною групою. У версії InterQual 2020 поряд з сотнями вдосконалень містяться також вказівки щодо догляду за пацієнтами з COVID-19.

InterQual AutoReview – це перше медичне рішення, яке автоматизує процес медичного огляду. Він використовує моделі обробки природної мови (NLP) для отримання даних з EHR та виявлення діагностичної інформації в неструктурованому клінічному вмісті.

1.5 Висновки до розділу 1 і постановка задачі дослідження

У ході виконання магістерської дисертації було виявлено, що проблема неправильно поставлених діагнозів є дуже недооціненою. У США вона є третьою серед причин смерті населення, а в Україні щодня від неї страждають близько 35 пацієнтів. Зважаючи на дані результати, було проведено аналіз виникнення та розвитку систем підтримки прийняття рішень у медицині. Робота над першими СППЛР почалася ще у 1970-х роках. На сьогоднішній день вони широко використовуються у багатьох країнах світу, особливо у США. Проте на разі більшість експертів надають перевагу СППЛР, що базуються на наукових знаннях, оскільки вважають їх більш надійними. Такі системи використовуються у різних сферах лікарської діяльності, від впорядкування медичних даних до визначення взаємодій різних препаратів. СППЛР, що не базуються на знаннях, використовують при діагностуванні, особливо для розпізнавання хвороб за набором симптомів та на основі графічних даних, таких як рентген, результати УЗД (ультразвукової діагностики) тощо. Для побудови таких систем широко використовують методи машинного навчання та нейронні мережі.

Постановка задачі:

1. Виконати аналіз актуальності проблеми, спеціальної літератури та існуючих результатів.

2. Зібрати необхідні статистичні дані щодо симптомів та наявності хвороби.

3. Побудувати діагностичну систему на основі байєсівської мережі.

4. Провести необхідні обчислювальні експерименти.

5. Проаналізувати отримані результати, оцінити ефективність роботи створеної системи.

РОЗДІЛ 2
ТЕОРЕТИЧНІ ОСНОВИ ТА ПРАКТИЧНЕ ЗАСТОСУВАННЯ БАЙЄСІВСЬКИХ МЕРЕЖ

У даному розділі розглядаються теоретичні основи байєсівських мереж, їх застосування у медичній діагностиці, а також деякі програмні продукти для побудови БМ.

2.1 Теоретичні основи байєсівських мереж

Термін “байєсівська мережа” (Bayesian Network) був запропонований американським вченим Джуді Перлом у 1985 році з метою узагальнення трьох аспектів [26]:

– об’єктивної природи вхідних даних;

– отримання достовірної інформації при застосуванні теореми Байєса;

– ідеї застосування аналізу причин та наслідків, запропонованої в 1763 році у посмертній роботі англійського священика Томаса Байєса.

Байєсівські мережі знаходяться на стику двох наук: теорії ймовірностей та теорії графів.

2.1.1 Теорема Байєса та основні поняття

Формально байєсівську мережу (БМ) можна описати як трійку [27]

$$N = < V,G,J >,$$

де V - множина змінних, що є вершинами графа G,

G - спрямований ациклічний граф, вузли якого відповідають випадковим змінним процесу, що моделюється,

J - спільний розподіл ймовірностей змінних $V = \{X_1, X_2, \dots, X_n\}$.

Стосовно множини змінних виконується марковська умова, тобто кожна змінна мережі не залежить від усіх інших змінних, за винятком батьківських попередників цієї змінної.

Таким чином БМ можна розглядати як модель представлення ймовірнісних залежностей (взаємозв'язків) між її вершинами. Зв'язок $A \rightarrow B$ називають причинним, якщо подія A є причиною виникнення B, тобто якщо існує механізм впливу значень змінної A на значення, які приймає змінна B. БМ називають причинною (каузальною) тоді, коли всі її зв'язки є причинними [27].

Отже, при побудові БМ спочатку необхідно визначити початкову множину змінних, що стануть її вершинами, далі обчислити значення взаємної інформації між ними. Потім необхідно знайти оптимальну структуру мережі. Як критерій якості можна використати оцінку опису мережі мінімальної довжини (ОМД).

Розглянемо теорему Байєса і формування висновку на її основі. Ймовірність одночасної появи двох незалежних подій D і S визначається за виразом:

$$p(D,S) = p(D)\, p(S).$$

Якщо події D і S залежні, то поява однієї з них дає деяку інформацію про можливість появи іншої:

$$p(D,S) = p(D)\, p(S \mid D)\ ,$$

де $p(S \mid D)$ - ймовірність появи події S за умови, що вже мала місце подія D.

У задачі медичної діагностики подію D можна інтерпретувати як хворобу, а S як її симптом. Якщо є інформація про те, що пацієнт має деяке захворювання, то можна присвоїти вищу ймовірність появи визначеного симптому. Враховуючи комутативність наведеного вище виразу, можна записати [27]:

$$p(D,S) = p(S)\, p(D \mid S) = p(D)\, p(S \mid D)\ .$$

Звідси маємо просту форму теореми Байєса (ТБ):

$$p(D \mid S) = \frac{p(D)\, p(S \mid D)}{p(S)}\ .$$

Теорему Байєса можна розглядати як механізм формування висновку (прийняття рішення). Розглянемо це на прикладі задачі постановки діагнозу.

Отже, маємо:

- p(D | S) – ймовірність захворювання при наявності у пацієнта симптому S, тобто це подія, відносно якої необхідно сформулювати висновок;
- p(D) – ймовірність захворювання на конкретну хворобу в межах деякої популяції (цю величину можна оцінити на основі аналізу історії розвитку цієї популяції);
- p(S | D) – ймовірність появи симптому, якщо пацієнт вже хворий (можна оцінити за допомогою даних, взятих з історій хвороб);
- p(S) – ймовірність появи даного симптому S у вибраній популяції (також можна обчислити на основі статистичних даних, але в цьому, як правило, немає необхідності (покажемо це нижче)).

Припустимо, що змінна захворювання D може приймати два можливих значення: D_t - істинне значення ймовірності, яке означає, що пацієнт має хворобу; D_f - неістинне (протилежне) значення. Ці два значення ймовірності дають в сумі одиницю незалежно від того, яке значення приймає S [27]:

$$\mathrm{p}(D_t \mid \mathrm{S}) + \mathrm{p}(D_f \mid \mathrm{S}) = 1.$$

Застосуємо до останньої рівності теорему Байєса:

$$\frac{\mathrm{p}(D_t)\ \mathrm{p}(\mathrm{S} \mid D_t)}{\mathrm{p}(\mathrm{S})} + \frac{\mathrm{p}(D_f)\ \mathrm{p}(\mathrm{S} \mid D_f)}{\mathrm{p}(\mathrm{S})} = 1$$

або

$$\mathrm{p}(\mathrm{S}) = \mathrm{p}(D_t)\ \mathrm{p}(\mathrm{S} \mid D_t) + \mathrm{p}(D_f)\ \mathrm{p}(\mathrm{S} \mid D_f).$$

Тобто знаючи оцінку p(S), її можна виключити з подальшого розгляду.

В даному прикладі змінна D має тільки два стани, але, очевидно, що p(S) можна виключити з розгляду при довільному числі станів D.

Теорему Байєса можна розглядати як вираз (механізм), який об'єднує «апріорну» та «правдоподібну» інформацію, запишемо її у вигляді:

$$p(D \mid S) = \alpha\, p(D)\, p(S \mid D),$$

де $\alpha = 1/ p(S)$ - нормуюча константа.

Тепер p(D) можна розглядати як апріорну інформацію, оскільки вона була відома до отримання будь-яких вимірів; p(S | D) - правдоподібна інформація, оскільки ми отримуємо її з аналізу (вимірів) симптомів.

В деяких випадках ми можемо обчислити апріорні ймовірності на основі статистичних даних [27]. Наприклад, апріорну ймовірність появи захворювання можна визначити в результаті ділення числа випадків захворювання на загальне число пацієнтів, які проходять огляд. Однак, в більшості випадків це неможливо зробити внаслідок суб'єктивних труднощів отримання статистичних даних, але апріорні знання можна представити у інших формах.

У випадку, коли дані щодо проблеми поступають з декількох джерел, теорема Байєса приймає вигляд [27]:

$$p(D \mid S_1, S_2, \ldots, S_n) = \frac{p(D)\, p(S_1, S_2, \ldots, S_n \mid D)}{p(S_1, S_2, \ldots, S_n)}.$$

В даному випадку виникає проблема оцінювання умовної ймовірності $p(S_1, S_2, \ldots, S_n \mid D)$ при великих значеннях n. Однак, якщо припустити незалежність подій S_i, $i = 1, \ldots, n$ при відомому D, то отримаємо:

$$p(S_1, S_2, \ldots, S_n \mid D) = p(S_1 \mid D)\, p(S_2 \mid D) \ldots p(S_n \mid D).$$

В результаті подальшого нормування можна позбутися знаменника p(S_1, S_2,..., S_n), що спрощує задачу формування висновку. Таким чином, отримуємо рівняння для формування висновку за теоремою Байєса:

$$p(D \mid S_1, S_2,..., S_n) = \alpha\ p(D)\ p(S_1 \mid D)\ p(S_2 \mid D)... p(S_n \mid D).$$

Це рівняння можна представити графічно (Рисунок 1) [27]. На графі змінні представлено колами, а стрілки вказують на зв'язок (умовні ймовірності) між незалежними і залежними змінними.

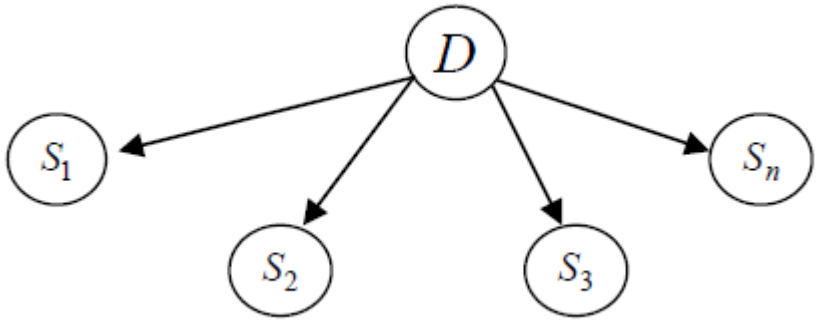

Рисунок 1 – Проста байєсівська мережа

Для того, щоб скористатись цією мережею, необхідно задати значення змінних, представлених вузлами. Надання значень вузлам (змінним) називають інстанціюванням.

Таким чином можемо бачити, що зручним та наочним способом представлення байєсівської мережі є графи з деякими специфічними властивостями. Як і будь-який граф, мережа Байєса складається з вузлів (змінних, вершин), з'єднаних дугами. Дуга, яка з'єднує змінні, вказує на існування причинного зв'язку між ними. Якщо дуга не спрямована, то вона свідчить тільки про наявність зв'язку між змінними. Графи, що містять тільки неспрямовані дуги, називають неспрямованими. Якщо дуга має стрілку, то вона вказує на напрям залежності між вершинами– від причини

до наслідку. Графи, у яких всі дуги спрямовані, називають спрямованими. А отже, всі байєсівські мережі – спрямовані графи.

Деякі змінні мереж Байєса мають специфічні назви [26]. Змінну (вузол, вершину) називають дочірньою (нащадком), якщо вона залежить від однієї або більше інших змінних. Незалежні змінні називають батьківськими або попередниками. Кожна батьківська змінна має одного або більше нащадків. До множини нащадків відносять похідні змінні, які відносяться до однієї батьківської змінної, а також дочірні змінні дочірніх змінних вищого рівня. Одна і та ж змінна може бути батьківською і дочірньою одночасно. Якщо змінна не має жодної батьківської, то її називають кореневою. Кореневі змінні – найважливіші змінні мережі; як правило, це атрибут, який досліджується. Змінну називають листом, якщо вона не має нащадків. Однозв'язний граф із змінними, що мають спеціальні назви, представлено на рисунку 2.

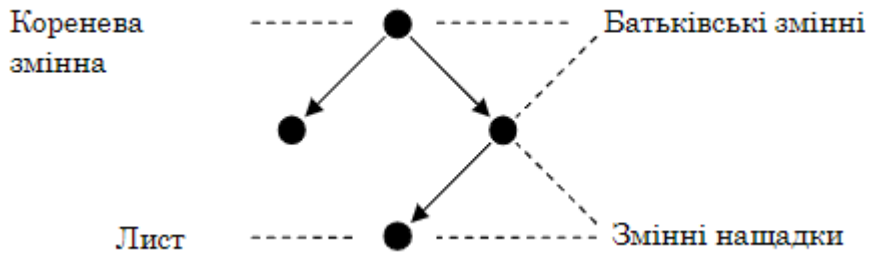

Рисунок 2 - Однозв'язний граф із спеціальними назвами змінних

Неоднозв'язність – це важлива властивість мережі. Якщо мережі однозв'язані, то висновок (рішення) можна знайти досить легко і прямолінійно. Але при вирішенні більшості практичних задач, як правило, мають справу з неоднозв'язними графами.

Існує декілька типів графічних структур байєсівської мережі [26]:

1. Дерево – структура, у якій будь-яка вершина може мати не більше однієї батьківської (Рисунок 3).

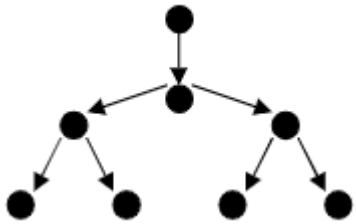

Рисунок 3 - Структура мережі Байєса у вигляді дерева

2. Полідерево – структура, у якій будь-яка вершина може мати більше однієї батьківської вершини, але при цьому між будь-якими двома вершинами повинно бути не більше одного з’єднуючого їх шляху (Рисунок 4).

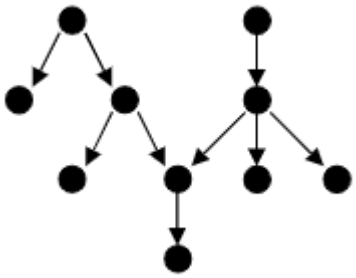

Рисунок 4 - Структура мережі Байєса у вигляді полідерева

3. Мережа – це структура, у якій будь-яка вершина може мати більше однієї батьківської вершини, при чому між будь-якими двома вершинами може бути більше одного з’єднуючого їх шляху (Рисунок 5).

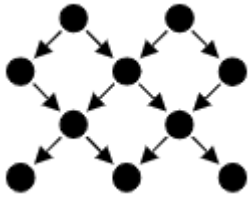

Рисунок 5 - Структура у вигляді мережі

Структури дерево та полідерево є однозв'язними мережами, а структури типу мережа – багатозв’язними мережами.

Якщо всі дуги графа спрямовані, то його називають спрямованим ациклічним графом. Якщо ж спрямований граф містить хоча б один замкнений цикл, то його називають спрямованим циклічним графом.

Приклади спрямованого ациклічного та циклічного графів показані на рисунку 6

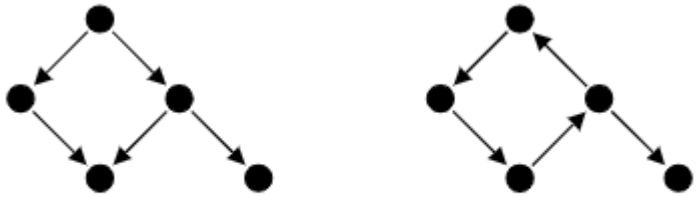

Рисунок 6 - Спрямовані ациклічний (а) та циклічний (б) графи

Спрямований ациклічний граф називають одиночно з'єднаним, якщо між будь-якими двома його змінними існує тільки один шлях, що їх з'єднує. Графи, показані на рисунку 6, не відносяться до такого типу. Очевидно, що графи з одиночними з'єднаннями (дерева та полідерева) є простішими для аналізу, але вони рідко зустрічаються при описанні реальних процесів та подій.

2.1.2 Методика побудови мережних імовірнісних моделей

Як було зазначено вище, байєсівські мережі представляються у вигляді спрямованого ациклічного графу (САГ) **G** на множині змінних X_i, $i = 1, \ldots, n$, що виступають його вершинами. Спрямовані дуги, що з'єднують вершини, вказують на існуючі залежності між змінними. Дочірні вузлові змінні описують таблицями умовного розподілу ймовірностей станів цих змінних за умови визначених станів батьківських змінних.

Спільний розподіл ймовірностей станів змінних САГ визначається за виразом [28, 29]:

$$p(X_1 = x_1, X_2 = x_2,..., X_n = x_n \mid G) = p(x_1, x_2, \dots, x_n) = \prod_{i=1}^{n} p(x_i \mid \boldsymbol{X}_{pa(i)}, G),$$

де G – спрямований ациклічний граф;

X_i, $i = 1, \dots, n$ – змінні (вершини) G;

x_i, $i = 1, \dots, n$ – можливі значення змінних X_i, $i = 1, \dots, n$;

$p(x_1, x_2,..., x_n \mid G)$ - ймовірність конкретної комбінації значень $x_1, x_2,..., x_n$ для множини змінних X_i, $i = 1, \dots, n$;

$\boldsymbol{X}_{pa(i)}$ - вектор безпосередніх батьківських змінних для Xi.

Умовні розподіли ймовірностей для дискретних змінних представляються множиною відповідних (багатовимірних) таблиць з параметрами

$$\Theta = \{\theta_1, \theta_2, \dots, \theta_n\} = \left\{\theta_{ik}(j)\big|_{k=1}^{r_i}\right\}_{j=1}^{q_i},$$

де $i = 1, \dots, n$ номери змінних $X_i \in$ X;

$k = 1, \dots, r_i$ – індекс, який вказує на значення змінної X_i;

$j = 1,..., q_i$ – індекс, що вказує на множину припустимих комбінацій значень батьківських змінних для X_i.

Розглянемо задачу побудови мережної моделі на основі вибірки даних потужністю N значень [29]. Позначимо через D = $\{ x^{(1)}, x^{(2)}, ..., x^{(n)} \}$ множину векторів даних, сформованих із значень станів $x_1^{(k)}, x_2^{(k)}, ..., x_n^{(k)}$ змінних X_i, $i = 1,..., n$. При цьому можливі такі випадки:

– структура БМ відома, потрібно оцінити лише її параметри;

– необхідно оцінити і структуру, і параметри мережі.

У разі наявності прихованих вершин або некоректності та / або неповноти даних потрібно застосувавати спеціальні підходи до навчання. Тому виділяють чотири випадки навчання БМ, які наведено у таблиці 1.

Таблиця 1 – Випадки навчання БМ

Структура	Спостереження	Метод
Відома	Повні	Метод максимальної правдоподібності
Відома	Часткові	Градієнтні методи, ЕМ-алгоритм (максимізація математичного сподівання), застосування генерування вибірки за Гіббсом
Невідома	Повні	Пошук в просторі моделей
Невідома	Часткові	Структурний ЕМ-алгоритм, алгоритм стиснення границь

Якщо структура мережі відома, а спостереження повні, для оцінювання параметрів мережі можна скористатись методом максимального апостеріорного оцінювання (МАО). Для дискретних змінних такі оцінки представляють собою відносні частоти появи кожного значення для кожної змінної при заданій конфігурації батьківських вузлів. Для реалізації процедури оцінювання за методом МАО необхідно вибрати апріорний розподіл для параметрів. З цією метою часто використовують розподіл Діріхле, спряжений стосовно багатовимірного розподілу функції правдоподібності.

У випадку, коли структура мережі невідома, необхідно спочатку оцінити структуру графа G, що включає у себе створення специфікацій стосовно умовної незалежності між змінними моделі і параметрами Θ [29]. Для оцінювання структури мережі на основі даних існує два основних підходи:

– оптимізаційні методи з урахуванням обмежень;
– пошукові методи на основі скорингових функцій.

Методи, що ґрунтуються на обмеженнях, досить ефективні, але їм бракує практичної робастності, тобто отримані в результаті структури дуже чутливі до помилок стосовно статистичного тестування умовної незалежності. Таким чином, у більшості випадків на практиці використовують пошукові алгоритми на основі скорингових функцій [29].

Розроблені на даний час процедури евристичного пошуку дають можливість знайти кращі структури мереж і зв'язані з ними розподіли ймовірностей у просторі усіх можливих структур БМ. Серед скорингових функцій популярні такі:

- байєсівська (ґрунтується на апостеріорній ймовірності графа **G**);
- апроксимації апостеріорних розподілів ймовірностей (ґрунтуються на інформаційному критерії Байєса);
- скорингова функція на основі опису мінімальної довжини (ОМД);
- інформаційно-геометричний критерій (info-geo) (враховує об'єм згортки, що відображає відповідну статистичну модель).

Методика побудови мережі на основі даних складається з таких кроків [29]:

1. Аналіз досліджуваного процесу (об'єкта) і скорочення розмірності задачі моделювання.
2. Масштабування і дискретизація змінних.
3. Визначення семантичних обмежень.
4. Оцінювання мережних моделей-кандидатів.
5. Аналіз якості і вибір кращої з моделей-кандидатів, застосування вибраної моделі для розв'язання поставленої задачі.

1. Скорочення розмірності задачі моделювання суттєво спрощує її подальше розв'язання, оскільки у більшості випадків при зростанні кількості змінних і параметрів кількість випадків (сесій) оцінювання цих змінних і параметрів зростає експоненційно [29]. Також редукція розмірності моделі сприяє підвищенню точності оцінок параметрів, оскільки скінченна вибірка

даних містить обмежений об’єм інформації. Для розв’язання задачі редукції можна використати такі методи:

- метод головних компонентів (МГК);
- факторний аналіз;
- багатовимірне шкалювання (БВШ);
- методи навчання на нелінійних структурах (наприклад локальне лінійне занурення).

Факторний аналіз, МГК і БВШ ґрунтуються на використанні власних векторів. За МГК обчислюються лінійні проекції максимальної дисперсії, що визначаються за власними векторами коваріаційної матриці вимірів [29]. Факторний аналіз ґрунтується на виявленні та моделюванні кореляційної структури даних, виключаючи з розгляду випадкові варіації даних. МГК частіше використовують для редукції вимірів (кількості змінних), а факторний аналіз – для виявлення структурних взаємозв’язків між змінними. Метод БВШ забезпечує обчислення проекцій малої розмірності, які якнайкраще зберігають попарні відстані між значеннями вимірів. Методи навчання на нелінійних структурах (конфігураціях) застосовують до деяких типів даних високої розмірності (наприклад, для розпізнавання образів), які можуть утворювати явно виражені суттєві нелінійності, оскільки, зазвичай, МГК, факторний аналіз або БВШ не дають у таких випадках прийнятних результатів.

2. На цьому кроці здійснюється масштабування розподілів даних з метою їх приведення до зручної для подальшого використання форми і дискретизація неперервних змінних, оскільки більшість відомих алгоритмів оцінювання структури і параметрів ймовірнісних моделей, а також формування висновку на їх основі ґрунтуються на дискретних даних. «Нестандартні» розподіли, наприклад розподіли з явно вираженою асиметрією, масштабують шляхом логарифмування або перетворення за методом квадратного кореня з метою наближення до розподілів відомих форм [29]. Очевидно, що при цьому втрачається первісний масштаб даних,

що необхідно враховувати у подальшій інтерпретації результатів оцінювання структур і параметрів моделей. Для дискретизації даних розроблено декілька ефективних схем, які забезпечують дотримання раціональних інтервалів у процесі дискретизації (наприклад, інтервали однакової ширини або однакових частот попадання значень). Розмір вибірки даних може накладати обмеження на кількість інтервалів, а також на кількість параметрів, які необхідно оцінити. Очевидно, що значення вибірки повинні бути представлені у кожному інтервалі. З одного боку, кількість інтервалів краще скорочувати, оскільки це дає можливість зменшити кількість оцінюваних параметрів. З іншого боку, скорочення розмірності даних призводить до зменшення їх роздільної здатності (точності представлення вимірів та експертних оцінок), а це в свою чергу зменшує точність зв'язаних з ними оцінок ймовірностей. Гранулярність (глибина) будь-якого аналізу визначається кількістю наявних варіантів подій та докладністю відповідних даних. Звідси випливає, що для поглибленого ситуаційного аналізу процесів та об'єктів довільної природи необхідно мати великі масиви даних, які забезпечують високу точність вимірів входів і виходів. Більшість алгоритмів оцінювання структури і параметрів мережних моделей дають кращі результати за умов відсутності пропущених значень, тобто відсутності інтервалів з пропущеними значеннями. Відсутність пропусків дає можливість застосовувати для оцінювання параметрів відносно простий метод максимальної правдоподібності, а не складний в реалізації метод максимізації математичного сподівання [29].

3. На цьому кроці здійснюється формулювання семантичних обмежень. У процедурі пошуку кращої структури мережі (як з повним, так і неповним перебором) необхідно задавати контекстно-спрямовані семантичні обмеження, які обмежують область пошуку структур. Оскільки розмірність простору пошуку експоненційно зростає при збільшенні кількості змінних моделі, то повний перебір практично неможливий [29]. Семантичні обмеження дають можливість здійснювати пошук тільки серед тих структур

мережі, які узгоджуються з часовими прецедентами або іншими вимогами залежності між змінними, що автоматично скорочує час, необхідний для обробки даних.

Прийнятне підґрунтя для формулювання семантичних обмежень надають базисна теорія каузальних структур і оцінки досвідчених експертів. У процесі побудови моделей необхідно, як мінімум, враховувати часові прецеденти взаємодії змінних між собою і при цьому не вносити значного зміщення у процес пошуку структури моделі. Семантичні обмеження сприяють скороченню кількості структур, які можна реалізувати, і підвищують ймовірність побудови раціональної структури. Раціональне використання знань стосовно предметної області (особливо при моделюванні об’єктів великої розмірності) дає можливість значно скоротити кількість можливих комбінацій вузлів, не знижуючи при цьому якість висновку, який формується на основі побудованої моделі [29].

4. На даному кроці здійснюється пошук структур моделей-кандидатів. За відповідними критеріями серед можливих структур моделей вибираються декілька кращих моделей-кандидатів та виконується оцінювання їх параметрів [29]. Методи побудови та оцінювання структури байєсівської мережі умовно розділяють на ті, що ґрунтуються на оціночних функціях (search & scoring) та ті, що працюють на основі тесту на умовну незалежність (dependency analysis). Детальніше розглянемо їх у окремому пункті.

Результатом використання кожної комбінації скорингової функції або тесту на умовну незалежність, алгоритму пошуку структури та відповідної вибірки даних є модель-кандидат, тобто мережа визначеної структури. Застосування комбінації скорингової функції та евристичного алгоритму перетворює задачу на оптимізаційну. Метою розв’язання цієї оптимізаційної задачі є оцінювання структури спрямованого ациклічного графа G у просторі допустимих структур Ω^{G}, який мінімізує значення скорингової функції і відповідає навчальним даним D [29].

У ролі скорингової функції можна використати апостеріорний розподіл ймовірностей

$$p(G \mid D, \Theta) \propto p(D \mid G)\, p(G),$$

але обчислення точних значень цієї функції навіть для мереж невеликої розмірності потребує значних обчислювальних витрат. Тому при оцінюванні розподілу p(G | D) роблять спрощення, наприклад, стосовно типу розподілу.

Для простої апроксимації апостеріорного розподілу ймовірностей мережі можна використати байєсівський інформаційний критерій (БІК), що є оцінкою маргінальної правдоподібності моделі на великих вибірках. Необхідно зазначити, що для отримання апроксимації прийнятної якості не потрібні великі вибірки, до того ж, у даному випадку не потрібно задавати апріорний розподіл для параметрів.

5. На цьому кроці здійснюється порівняння характеристик обраних моделей-кандидатів з подальшим вибором найкращої для опису досліджуваного процесу. Для оцінювання якості моделей такого типу застосовують критерії точності прогнозування з використанням наявних даних для тестування. Наприклад, щоб оцінити якість роботи моделі класифікації обчислюють усереднену зважену корисність (або вартість), отриману за допомогою її ймовірнісних прогнозів. Такий підхід можливий у випадках, коли можна отримати інформацію стосовно вартості можливих втрат від некоректної класифікації або корисності, досягнутої завдяки правильній обробці даних [29]. Ще однією метрикою для порівняння істинного спільного розподілу ймовірностей (він завжди невідомий) з його оцінкою є відстань Кульбака-Лейблера, яку можна розглядати як деяку стандартизовану оцінку якості побудованої моделі, у тому числі байєсівської мережі.

Якщо нам відома оригінальна байєсівська мережа, тобто та, за якою генерувалися дані, то для оцінювання якості побудови моделей-кандидатів

можна використовувати облік кількості зайвих, відсутніх і реверсованих дуг у побудованій БМ порівняно з оригінальною [26]. Мірою оцінювання похибки побудови може бути структурна різниця (structure difference) або перехресна ентропія (cross entropy) між побудованою і оригінальною мережею Байєса.

Структурна різниця

Для обчислення структурної різниці використовують формулу симетричної різниці структур [26]:

$$\delta = \sum_{i=1}^{n} \delta_i = \sum_{i=1}^{n} card((\Pi^{(i)}(B) \backslash \Pi^{(i)}(A)) \cup (\Pi^{(i)}(A) \backslash \Pi^{(i)}(B)),$$

де B – побудована БМ;

A– оригінальна БМ;

n– кількість вершин мережі;

$\Pi^{(i)}(B)$ - множина предків i–ї вершини побудованої мережі B;

$\Pi^{(i)}(A)$ – множина предків i–ї вершини оригінальної мережі A;

card(ξ) – потужність скінченої множини ξ, що визначається кількістю елементів, які їй належать.

Перехресна ентропія

Перехресна ентропія – це відстань між розподілом побудованої і оригінальної БМ [26]:

$$H(p,q) = \sum_{v} p(v) \cdot \log \frac{p(v)}{q(v)} =$$

$$= \sum_{j \in J} \sum_{s \in S(j,g)} \sum_{a \in A^{(j)}} p\left(X^{(j)} = a \middle| \Pi^{(j)} = s\right) \cdot \log \frac{p\left(X^{(j)} = a \middle| \Pi^{(j)} = s\right)}{q\left(X^{(j)} = a \middle| \Pi^{(j)} = s\right)},$$

де $p(v)$ – спільний розподіл оригінальної БМ;

$q(v)$ – спільний розподіл побудованої БМ.

2.1.3 Методи побудови та оцінювання структури байєсівської мережі

Якщо структура байєсівської мережі, що моделює досліджуваний процес, не відома, але є дані, то для її побудови та оцінювання можна скористатися методами на основі скорингових функцій або оптимізаційними методами з урахуванням обмежень та тестів на умовну незалежність. Розглянемо деякі з них.

PC (path condition) алгоритм. PC алгоритм, розроблений в 1991 році спеціально для побудови розріджених (sparse) БМ, тобто для мереж із невеликою кількістю дуг між вершинами. Алгоритм HUGIN PC [30], який є варіантом оригінального PC алгоритму належить до класу алгоритмів навчання на основі обмежень. Основна ідея цих алгоритмів полягає у формуванні набору тверджень про умовну незалежність та залежність (CIDS, conditional independence and dependence statements) за допомогою статистичних тестів.

Алгоритм виконує наступні кроки:

1. Статистичні тести на умовну незалежність виконуються для всіх пар змінних (за винятком пар, для яких було вказано структурне обмеження).
2. Між кожною парою змінних, для яких не виявлено умовної незалежності додається неорієнтоване ребро. Отриманий неорієнтований граф називають скелетом мережі.
3. Потім ідентифікуються коллайдери (пара посилань, спрямованих таким чином, що вони зустрічаються у вузлі), що гарантують відсутність спрямованих циклів. Наприклад, якщо ми виявимо, що між A і B існує залежність, між B і C є залежність, але A і C є

умовно незалежними з урахуванням S, що не містить B, тоді це може бути представлено структурою A -> B <- C.

4. Далі встановлюються напрямки для тих зв’язків, напрямок яких можна отримати із знайдених умовних незалежностей та ідентифікованих колайдерів.
5. Решта неспрямованих зв’язків направляються випадковим чином, за умови відсутності спрямованих циклів.

Таким чином можемо зауважити, що зазвичай PC алгоритм не може визначити напрямок усіх зв’язків із даних, а це означає, що деякі посилання будуть спрямовані випадковим чином. Тому отриману мережу потрібно перевірити на наявність хибно направлених зв’язків.

NPC (Necessary Path Condition) алгоритм. NPC (Necessary Path Condition, умова необхідного шляху) алгоритм був розроблений дослідниками з Siemens в Мюнхені для вирішення деяких проблем алгоритмів навчання на основі обмежень, таких як алгоритм PC [30]. Обидва алгоритми PC та NPC базуються на формуванні скелета мережі, виведеного через статистичні тести на умовну незалежність. Алгоритм NPC створений для виправлення недоліків алгоритму PC, які виникають особливо в умовах малих наборів даних. Здійснюється це за рахунок додавання умови необхідного шляху, що є основою для введення поняття неоднозначних областей, які містять набори взаємозалежних невизначених зв’язків. Рішення щодо напрямку таких зв’язків приймає користувач.

Умова необхідного шляху

Загалом умова необхідного шляху означає, що для того, щоб дві змінні X і Y були умовно незалежними на множині S, без належної підмножини S, для якої це справедливо, повинен існувати шлях між X і кожним Z із S (що не перетинаються з Y) та між Y та кожним Z із S (що не перетинаються з X). В іншому випадку включення Z у S є незрозумілим. Таким чином, для того, щоб існувала незалежність, на графіку має бути присутнім ряд посилань.

Неоднозначні області

Коли відсутність зв’язку *a*, залежить від наявності іншого зв’язку *b*, і навпаки, ми визначаємо *a* і *b* як взаємозалежні. Таким чином *a* і *b* є невизначеними зв'язками. Неоднозначна область - це максимальна множина взаємозалежних зв’язків. Головна мета - отримати якомога менше якомога менших неоднозначних областей. Слід зазначити, що детерміновані зв’язки між змінними також можуть створювати неоднозначні області. Якщо є якісь невизначені або ненаправлені зв’язки, користувач може надати додаткову інформацію про них для вирішення неоднозначних областей.

Greedy search-and-score algorithm. Жадібний алгоритм пошуку та оцінки (greedy search-and-score algorithm) належить до алгоритмів на основі скорингових функцій. Алгоритм виконує пошук у просторі можливих мережевих структур і повертає структуру з найвищим балом. Зміна структури мережі-кандидата відбувається за рахунок додавання та видалення дуг або зміни їх напрямку. Для кожної мережі-кандидата розраховується оцінка максимальної правдоподібності умовних таблиць ймовірності, пов'язаних з вузлами мережі, для обчислення балу структури кандидата. Якщо дані неповні, слід використовувати алгоритм EM. Однак, якщо дані повні, це можна зробити шляхом підрахунку частоти випадків та її нормалізації [31]. У якості скорингової функції можна обрати інформаційний критерій Акайке (AIC) або інформаційний критерій Байєса (BIC).

Chow-Liu Tree. Дерево Чу-Ліу (Chow-Liu Tree) найкраще підходить для пошуку структур типу дерево, але не підходить для багатоз’язних БМ. Якість наближення вимірюється за допомогою відстані Кульбака-Лейблера між справжнім розподілом та знайденим [32]. Якщо навчання відбувається за даними, то справжній розподіл визначається частотою спостережень. Даний алгоритм Чу та Ліу представили у 1968 році. Вони показали, що оптимальним буде дерево, яке включає максимальну вагу всіх змінних, при чому вага кожного ребра подається як взаємна інформація між змінними, пов'язаними цим ребром.

Алгоритм побудови дерева Чу та Ліу:

1. Обчислюється взаємна інформація MI(X_i, X_j) для кожної пари (X_i, X_j). Отримуємо повний неорієнтований граф, де кожен зв'язок між X_i та X_j має вагу MI(X_i, X_j).
2. Будується дерево, що включає максимальну вагу всіх змінних отриманого повного неорієнтованого графу.
3. Певна змінна отриманого дерева обирається у ролі кореня. Всі зв'язки направляються у протилежному напрямку від нього.

Rebane-Pearl Polytree. Алгоритм полідерева Рібана-Перла (Rebane-Pearl Polytree) був створений у 1988 році. Він є удосконаленою модифікацією дерева Чу-Ліу. Замість побудови структури у вигляді дерева, де кожен вузол має лише один батьківський, крім кореневого, алгоритм створює мережу у формі полідерева, де кожен вузол може мати декількох батьків, але основна мережева структура все ще є деревом.

Tree Augmented Naive Bayes. Наївний байєсівський класифікатор з додаванням дерева (TAN, Tree Augmented Naive Bayes) ґрунтується на алгоритмі Чу-Ліу. Алгоритм TAN корисний для побудови мереж класифікації, де конкретний вузол моделі є об'єктом міркувань. Цільовий вузол використовується для побудови умовного дерева Чу-Ліу (тобто дерева Чу-Ліу, що оперує усіма вузлами, крім вибраного цільового) з вибраним коренем як коренем дерева. Ваги визначаються як умовна взаємна інформація (залежна від цільового вузла), а всі вузли (крім цільового) мають цільовий як додатковий батьківський вузол [33].

2.1.4 Алгоритм Hugin формування ймовірнісного висновку

Процес обчислення оцінки стану вершини на основі апріорної ймовірності про стани інших вершин мережі Байєса називають формуванням

(обчисленням, отриманням) ймовірнісного висновку. Саме механізм побудови ймовірнісного висновку перетворює будь-яку мережу Байєса, яка описує відповідний процес, на повноцінну складову експертної системи [26].

У 1990 році Ф. Дженсен, К. Олесен та С. Андерсен [34] створили метод Hugin, який являвся модифікацією LS-методу.

Ідея алгоритму Hugin полягає в тому, що для реалізації ймовірнісного висновку структура байєсівської мережі спочатку приводиться до вигляду об’єднаного дерева, а потім використовується алгоритм розповсюдження повідомлень по дереву догори та донизу і послідовно перераховуються таблиці умовних ймовірностей вершин дерева. У літературі, замість терміну “об’єднані дерева” (junction trees), іноді застосовуються терміни дерева клік (clique trees), гіпердерева (hypertrees) та якісні дерева Маркова (qualitative Markov trees) [вел. книжка].

Для реалізації алгоритму Hugin потрібно здійснити два етапи. На першому будується об’єднане дерево клік з первинної структури мережі, а потім його вершини заповнюються таблицями умовних ймовірностей. Даний етап реалізується у чотири кроки [26]:

1. Моралізація графа.

2. Триангуляція графа.

3. Ідентифікація клік.

4. Побудова об’єднаного дерева.

Розглянемо необхідні кроки детальніше.

1. Моралізація графа G означає, що при послідовному переборі всіх вершин БМ, у яких є батьки, якщо батьки вершини не зв’язані між собою, то між ними встановлюється зв'язок. Після цього всі спрямовані дуги графа замінюються неспрямованими.

2. Триангуляція моралізованого графу передбачає розбиття моралізованого графу на трикутники. Для цього застосовують алгоритм часової заміни (fill-in computation algorithm). Процес триангуляції можна описати таким чином:

1) послідовно перебираються всі вершини БМ таким чином [26]:

- перевіряється, чи є суміжними між собою сусідні вершини аналізованої; якщо так, то така вершина симпліціальна – утворює кліку разом із своїми сусідами (така вершина виключається з розгляду разом із її ребрами);
- якщо після перебору всіх вершин мережа, що залишилася до розгляду, не порожня, то перейти до наступного кроку алгоритму; у протилежному випадку, вважається, що граф триангульований;

2) у мережі, що залишилась до розгляду, послідовно перебираються вершини таким чином: шукається вершина з найбільшою кількістю сусідів (така вершина стає симпліціальною шляхом введення додаткових ребер між її несуміжними сусідами), тобто тепер вона утворює кліку з сусідами, а потім ця вершина виключається із розгляду [26]. Після розгляду усієї мережі до початкового моралізованого ненаправленого графа додаються додаткові ребра, і такий граф буде триангульованим.

3. За допомогою алгоритму пошуку клік (cliques-finding algorithm) в триангульованому моралізованому графі визначається множина клік, тобто підграфів, потужності яких не перевищують трьох.

Для поточної кліки виконуються такі дії [26]: перевіряється, чи є вона підмножиною інших не перебраних клік, якщо це так, то вона знищується; якщо в ній та інших кліках співпадає хоча б одна вершина, то між відповідними кліками вводиться ребро, що містить сепаратор, тобто перетин множин вершин цих клік. Після цього виконується ранжирування множини клік, застосовуючи упорядковану множину вершин.

Ранжирування множини клік робиться таким чином [26]: першою клікою стає та, що має вершину, яка в упорядкованій множині вершин посідає перше місце; якщо клік, які мають цю вершину є декілька, то потрібно звернути увагу на наявність у кліці вершини, яка займає друге місце

в упорядкованій множині вершин, і так далі. Таким чином виконується ранжирування усієї множини клік.

4. На цьому етапі будується об’єднане дерево. Під час реалізації даного кроку всі кліки послідовно зв’язуються між собою шляхом вибору для зв’язку ребер з найбільшими сепараторами, а інші ребра видаляються [26]. Дерево, що зв’язує всі кліки з максимальними потужностями сепараторів, буде об’єднаним деревом. Для зручності, як корінь в об’єднаному дереві, вибирається кліка з найбільшою кількістю вершин. Якщо таких вершин декілька, то перевага віддається кліці з найбільшою кількістю ребер.

На завершення першого етапу об’єднане дерево поетапно заповнюється таблицями значень. За отриманими значеннями ймовірностей клік, можна обчислити значення спільної ймовірності об’єднаного дерева [26]:

$$p(\text{мережі}) = p(\text{кліка}_1, \dots, \text{кліка}_n) = \prod_i \psi_i,$$

де ψ_i – значення імовірності i-тої кліки.

Другий етап (алгоритм пропагації) передбачає обчислення значень ймовірностей станів вершин на основі алгоритмів розповсюдження значень ймовірності по об’єднаному дереву. Для обчислення значень ймовірностей клік використовуються l і p – повідомлення [26]. Після цього за значеннями ймовірностей клік обчислюють індивідуальні ймовірності вершин.

Спочатку виконують процедуру "сходження догори" (upword). Повідомлення є результатом маргіналізації – підсумовування змінних таблиць, які не містяться в сепараторі. Після відсилання повідомлення відправник ділить свою поточну таблицю умовних ймовірностей на нього. Коли отримувачу надходить повідомлення, він множить його на свою таблицю умовних ймовірностей і виходить нова таблиця [26].

Коли він отримує повідомлення від усіх своїх нащадків, то, у свою чергу, посилає також повідомлення своєму батьку і ділить свою таблицю на

відіслане повідомлення. Процес триває доти, доки корінь зв’язного дерева не отримає повідомлення від усіх своїх нащадків.

Потім виконують процедуру "сходження донизу" (downward). Корінь посилає повідомлення кожному своєму нащадкові. Він ділить свою таблицю умовних ймовірностей на повідомлення, отримане від нащадка, маргіналізує таблицю по сепаратору і посилає результат. Коли нащадок отримує повідомлення від свого батька, він множить його на свою поточну таблицю умовних ймовірностей і формує таким чином свою нову таблицю. Далі він маргіналізує її (по сепаратору) і посилає своєму нащадкові. Процес триває доти, доки усі листкові вершини не отримають повідомлення. Таким чином, результатом будуть знову перераховані таблиці для кожної змінної за умови наявності спостережень, які потім необхідно нормалізувати [26].

Знизу нагору йдуть l- повідомлення, а потім зверху вниз йдуть p - повідомлення, у процесі проходження повідомлень відбувається перерахування значення ймовірності кліки ψ_i.

Далі для кожної вершини виконується пошук клік, у яких вона міститься. Якщо серед них є кліка, яка не є листом, то вибирається ця кліка. В іншому випадку – будь-яка з них. Щоб отримати ймовірність кожного стану вершини, треба додати усі значення ймовірностей для цього конкретного стану, які є в таблиці кліки. За значенням ймовірності кліки виконується обчислення ймовірності кожної вершини кліки:

$$p(R_i|S_i) = \frac{p(R_i,S_i)}{p(S_i)}.$$

Це рівняння застосовується для обчислення значень ймовірностей інстанційованих вершин.

Детальніше даний метод можна розглянути у [26].

2.2 Огляд застосування байєсівських мереж у медичній діагностиці

Застосування байєсівських мереж при постановці медичних діагнозів розпочалось у 1980-х роках. Одним із перших досягнень стала QMR-система (Quick Medical Reference, Швидка медична довідка), розроблена у 1980 році в Пітсбургському університеті. Сьогодні мережа Байєса QMR-системи складається приблизно з 6000 вершин, з'єднаних більш ніж 415000 дугами. Система спроможна розпізнати близько 750 видів різних захворювань за більше ніж 5000 симптомами та результатами лабораторних аналізів.

Першою експертною системою на основі байєсівських мереж була CONVINCE. Вона складається із статистичних та експертних даних, які використовуються при діагностиці для допомоги лікарю-терапевту.

Однією з перших медичних діагностичних систем є PATHFINDER, розроблена для діагностики захворювань лімфовузлів, що реалізує 60 різноманітних варіантів постановки діагнозу та має 130 змінних-симптомів, значення яких можуть спостерігатися при вивченні клінічних випадків. Якість діагностики, забезпечувана системою наблизилася до рівня експертів, що сприяло зростанню її популярності [26].

Згадаємо також й деякі інші медичні експертні системи [26], зокрема NESTOR – систему діагностування ендокринологічних порушень, MUNIN – систему діагностування нервомускулярних порушень, PATHFINDER IV – систему діагностування захворювань лімфатичної системи, DIAVAL – експертну систему для ехокардіографії, а також розробки компанії Knowledge Industries, Inc. (KI), які охоплюють діагностування та лікування порушень сну, психічні патології, травматологію, аналіз стану руки і зап'ястя, дерматологію та оцінку здоров'я в домашніх умовах, Hepar II –

систему діагностування захворювань печінки, а також системи Child, Munin, Painulium, SWAN.

Розглянемо системи Hepar II та PATHFINDER детальніше.

2.2.1 Hepar II

Hepar II створеної для зменшення кількості процедур біопсії печінки [26]. Вона сконструйована за гібридним підходом: структура мережі побудована на основі знань експертів і відповідній медичній літературі, а числові дані для моделі, тобто ймовірнісний розподіл для вершин – з бази даних проб печінки Hepar, яка була створена у 1990. Байєсівська мережа системи Hepar II (Рисунок 7) має близько 70 вершин, 11 з них – це різні хвороби печінки, а 61 – дані, що надав пацієнт, ознаки, симптоми хвороб і результати лабораторних тестів. Загалом система містить 1488 числових параметрів.

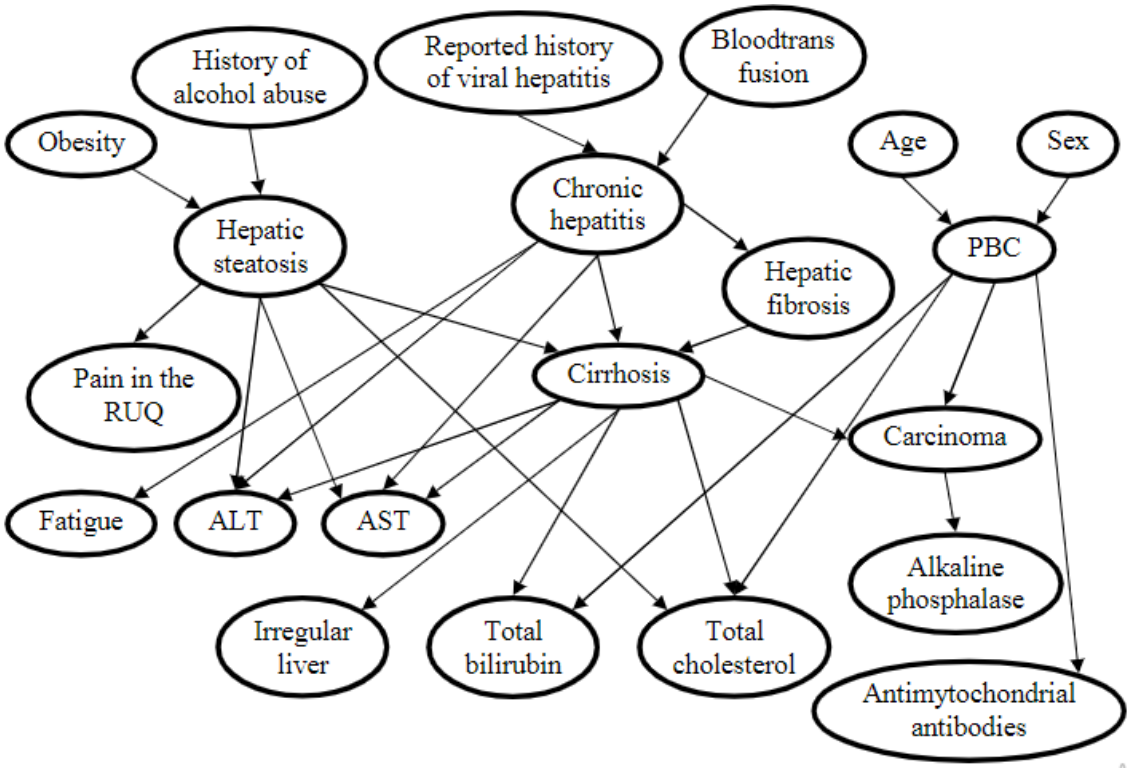

Рисунок 7 – Спрощений фрагмент байєсівської мережі Hepar II

Дослідження ефективності роботи системи відбувалось за участі 23 медичних фахівців. Результат першого тесту, що полягав у перевірці діагнозу десяти випадково вибраних із бази даних пацієнтів, показав, що діагнози, поставлені за допомогою HeparII були вірними у 70% випадків, у той час як визначені фахівціями-медиками лише у 33,1%. Іншим важливим результатом дослідження виявилось те, що завдяки правильному висновку системи, лікарі коригували своє рішення, і таким чином точність поставлених діагнозів збільшилася з 33,1 до 65,8 %.

У 2001 році результати порівняння Hepar II з Hepar-RB (системою для діагностування захворювань печінки на множині правил), показало, що обидва підходи є ефективними, хоча і мають свої переваги та недоліки [26]. Підхід на базі правил дозволяє протестувати модель шляхом послідовної перевірки прийнятих системою рішень, що можливо і для мереж Байєса, якщо реалізувати автоматичне генерування пояснень. Важливою особливістю байєсівських мереж є можливість отримання структури моделі з бази даних. Системи ж на базі правил краще застосовувати у випадках, коли причина-наслідок – не основний принцип, закладений у задачу або ж коли вона занадто складна, щоб бути представленою каузальною мережею. Крім того, як підтвердили експерименти, система на основі множини правил має труднощі з обробкою неповних даних – в системі Hepar-RB близько 35% пацієнтів не отримали діагнозу, тоді як в Hepar II ця цифра становила лише 2%.

2.2.2 PATHFINDER

Перейдемо до PATHFINDER. Вона призначена для діагностики захворювань лімфовузлів [26].

PATHFINDER включає 60 різноманітних варіантів діагнозу та 130 змінних симптомів (Рисунок 8), значення яких можуть спостерігатися при вивченні клінічних випадків.

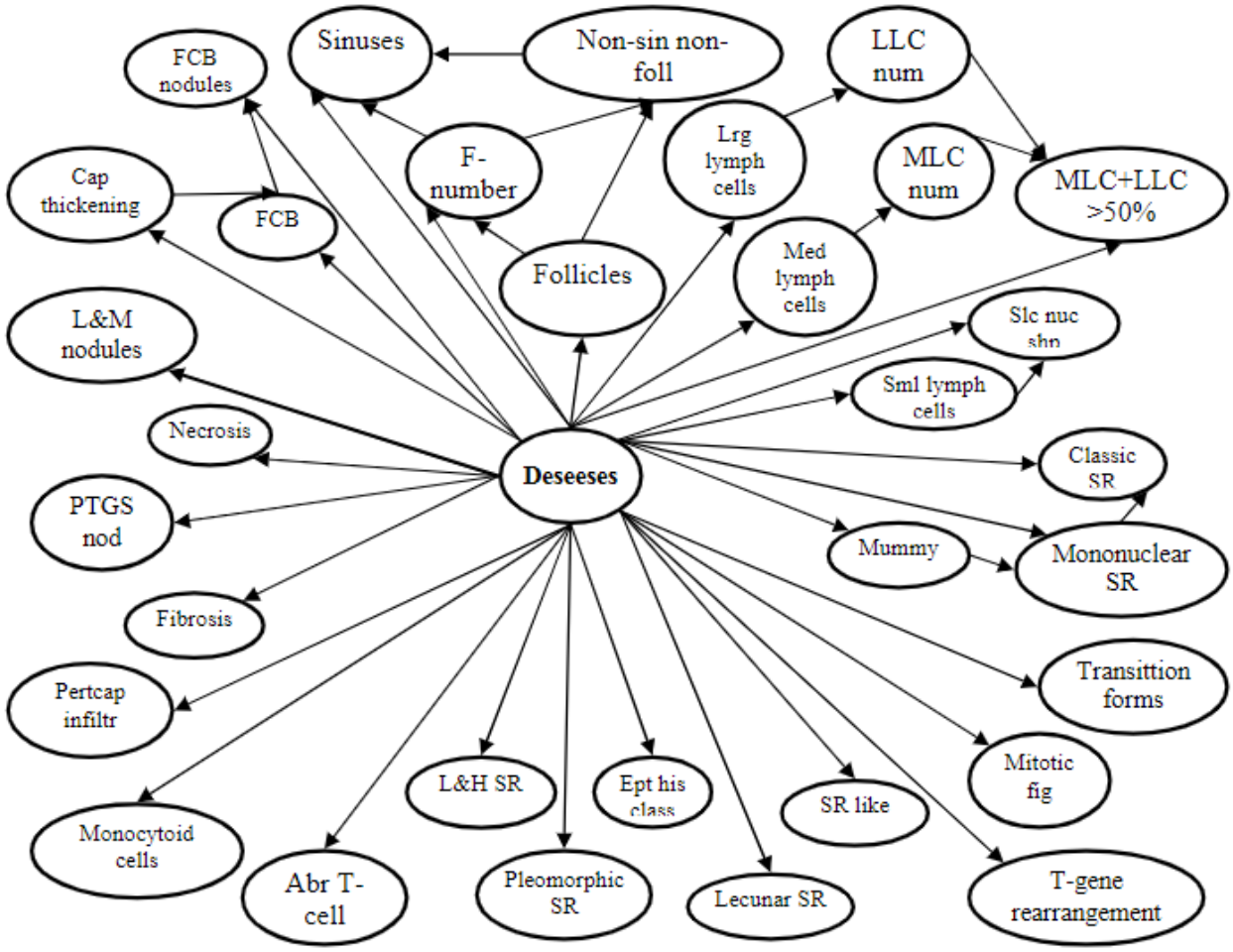

Рисунок 8 – фрагмет байєсівської мережі PATHFINDER

Система змогла наблизитися до рівня експертів, і її четверта версія набула стану комерційної системи – Intellipath.

2.3 Деякі програмні продукти для побудови байєсівських мереж

Сьогодні розроблено та активно використовується значна кількість різноманітних програмних продуктів, що реалізують байєсівські мережі. Розглянемо деякі з них.

2.3.1 AgenaRisk

Компанія Agena з'явилася у 1998 році, а систему AgenaRisk було випущено у 2003 році.

AgenaRisk [35] надає програмне забезпечення для побудови байєсівських мереж для аналізу ризиків, застосування методів штучного інтелекту та додатків для прийняття рішень. AgenaRisk використовує найновіші розробки в галузі байєсівського штучного інтелекту та ймовірнісних міркувань для моделювання складних, ризикованих проблем та вдосконалення способу прийняття рішень. За допомогою моделей AgenaRisk можна здійснювати прогнозування, проведення діагностики та прийняття рішень, поєднуючи дані та знання про складні причинно-наслідкові та інші залежності в реальному світі (Рисунок 9).

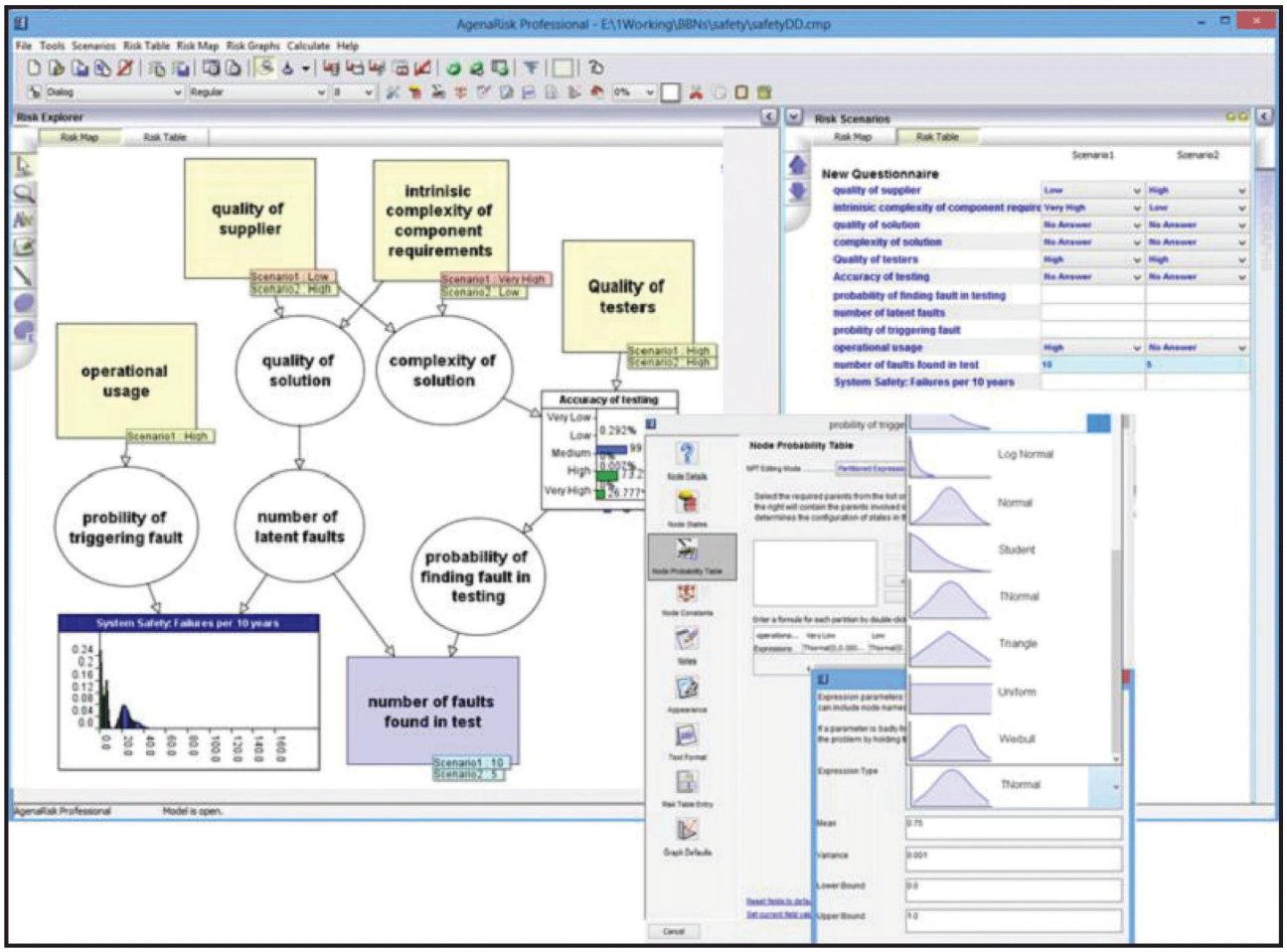

Рисунок 9 – Інтерфейс AgenaRisk Professional

AgenaRisk – комерційний проект (вартість ліцензії від 2100£ на рік). Для працівників та студентів акредитованих вищих навчальних закладів доступна академічна ліцензія за зниженою ціною від 525£. Також у даного програмного забезпечення є безкоштовна пробна версія, якою можна користуватися протягом 14 днів, а далі з обмеженими функціональними можливостями і лише вбудованими моделями.

2.3.2 BayesiaLab

Bayesia S.A.S. – це французька компанія з розробки програмного забезпечення, заснована в 2001 році докторами Лайонелом Жуффом (Dr.

Lionel Jouffe) та Полом Мунтеану (Dr. Paul Munteanu), яка спеціалізується на технологіях штучного інтелекту.

Портфель програмного забезпечення Bayesia зосереджений на всіх аспектах підтримки прийняття рішень за допомогою байєсівських мереж і включає BayesiaLab (Рисунок 10), BEST та BRICKS [36]. Спектр їх застосування варіюється від індивідуальної підтримки прийняття рішень до масштабного аналізу політики та оцінки ризиків промислових систем.

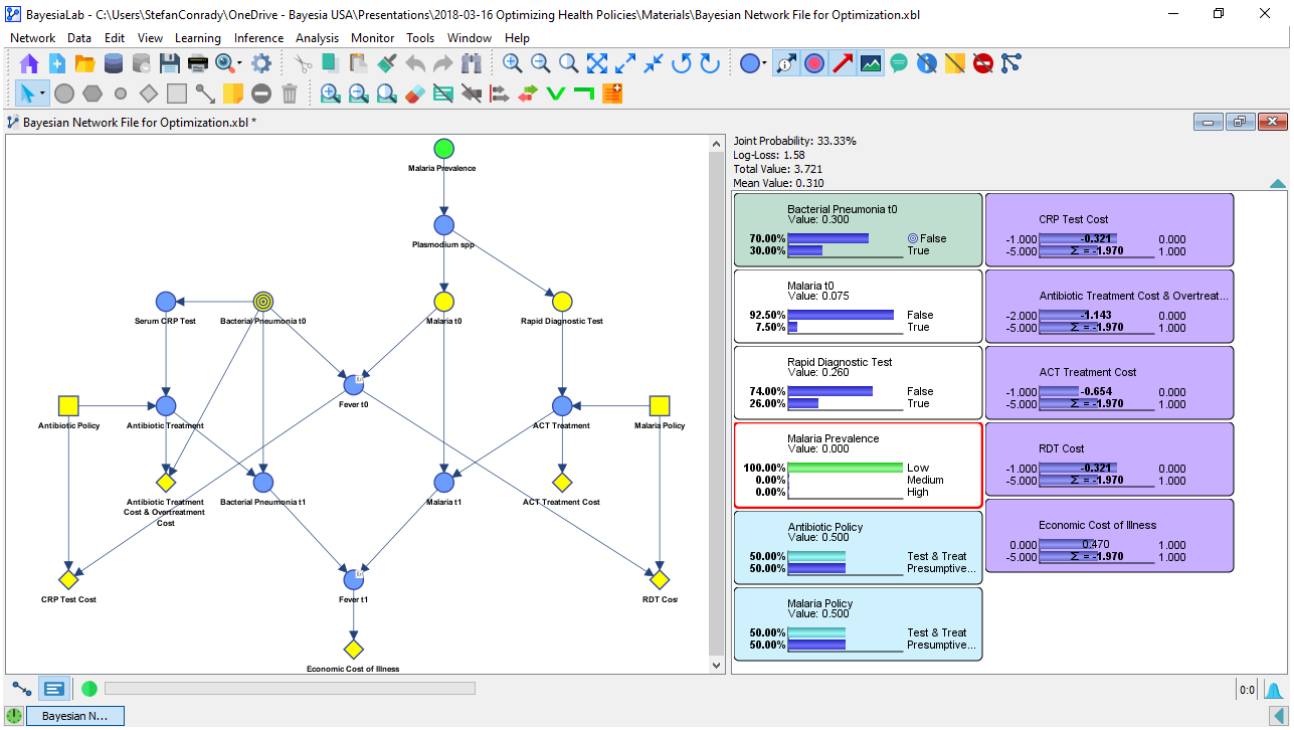

Рисунок 10 – Інтерфейс програми BayesiaLab

Щодо вартості ліцензії, то вона варіюється від 590€ на рік за некомерційну академічну версію до 99990€ за комерційну професійну.

2.3.3 Bayes Server

Bayes Server [37] використовує байєсівські мережі для таких завдань, як класифікація, регресійний аналіз, прогнозування часових рядів, сегментація / кластеризація, оцінка щільності, виявлення аномалій, підтримка прийняття рішень, аналіз багатовимірних даних і ін. Дозволяє працювати як з призначеним для користувача інтерфейсом, так і з кросплатформним API (Рисунок 11). Перша версія вийшла в 2008 році, але до цього розробки велися протягом десяти років [38].

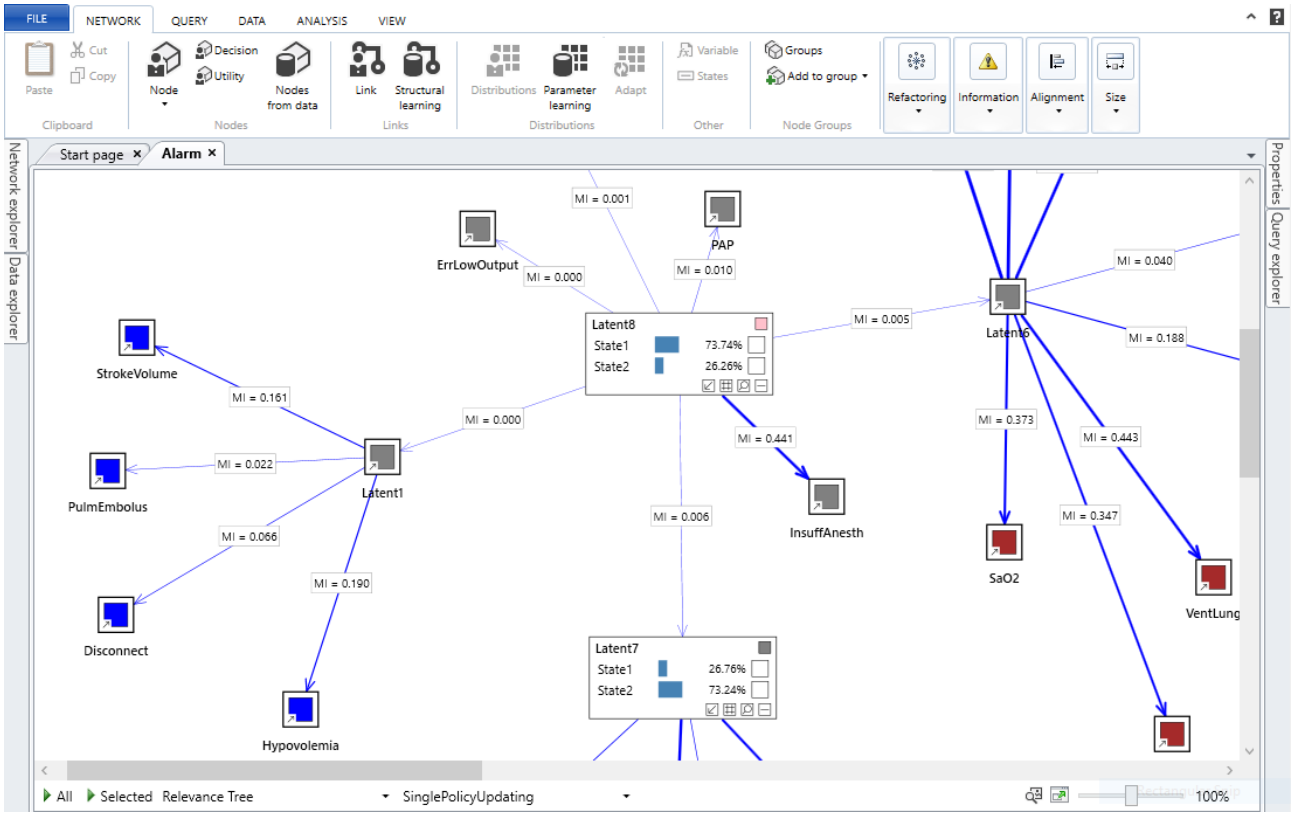

Рисунок 11– Інтерфейс програми Bayes Server

Вартість ліцензії варіюється від 648,70$ за некомерційну академічну версію до 1755$ за комерційну професійну.

2.3.4 GeNIe

GeNIe [39], який є ще одним із безкоштовних програмних продуктів, розроблена лабораторією систем підтримки прийняття рішень Пітсбургського Університету США. Програмне забезпечення GeNIe дуже зручне у використанні, надає можливість автоматичної побудови мереж Байєса та підключення власних модулів з алгоритмами, написаними на мові програмування С.

GeNIe Modeler був розроблений у Піттсбургському університеті в 1998 році, з 2015 року ліцензія на цей продукт належить компанії BayesFusion. GeNIe Modeler («Graphical Network Interface» – графічний інтерфейс мережі) – це пакет програмного забезпечення, який може бути використаний для створення теоретичних моделей (Рисунок 12).

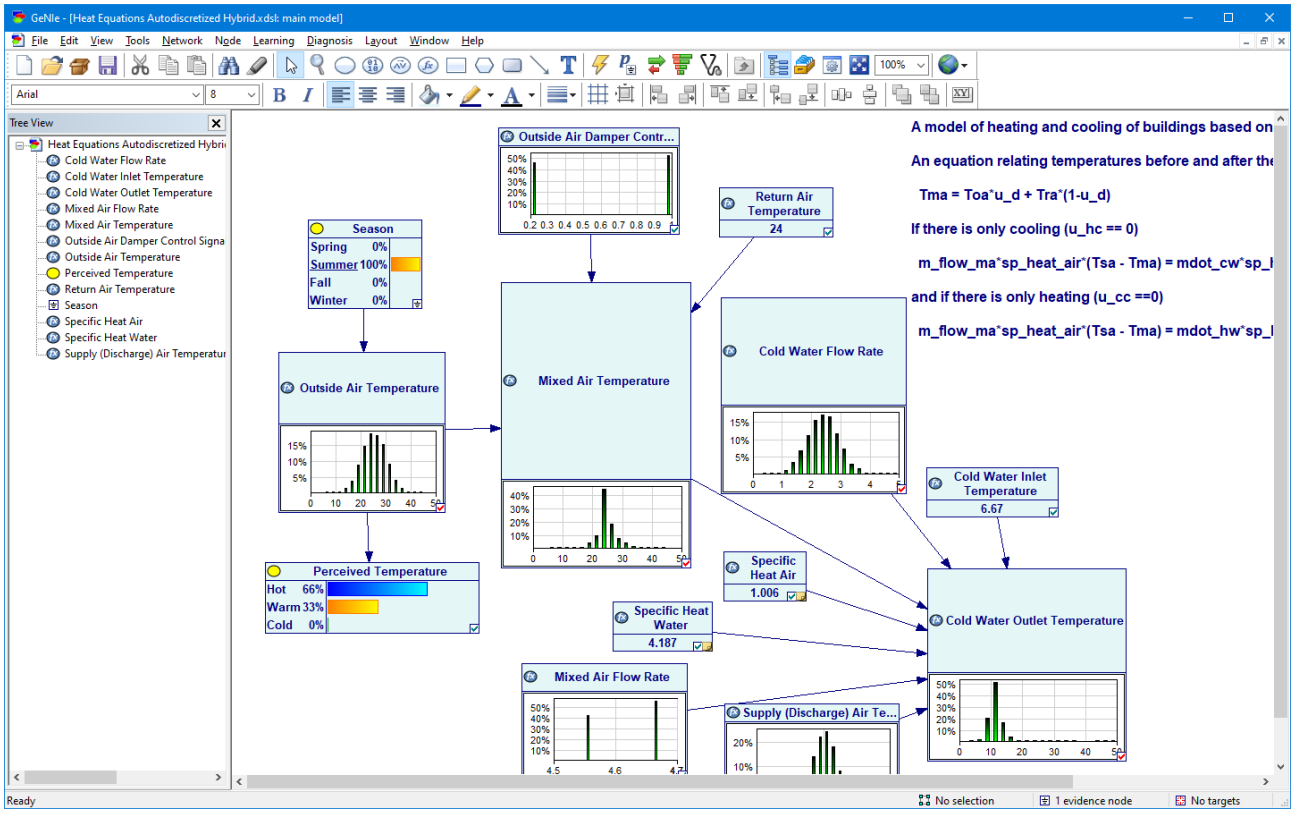

Рисунок 12– Інтерфейс GeNIe Modeler

GeNIe Modeler є графічним інтерфейсом для SMILE [26]. SMILE («Structural Modeling, Inference, and Learning Engine» – знаряддя для структурного моделювання, виведення і навчання) є повністю незалежною від платформи бібліотекою класів C ++, що реалізує ймовірнісно-графічні моделі і моделі рішень, такі як байєсівські мережі, діаграми впливу і моделі структурних рівнянь. Його окремі класи, визначені в SMILE API, дозволяють створювати, редагувати, зберігати і завантажувати графічні моделі і використовувати їх для імовірнісного виведення і прийняття рішень в умовах невизначеності. SMILE підтримує методологію об'єктно-орієнтованого програмування. Окремі класи SMILE доступні з C ++ або (як функції) з C. Завдяки реалізації на C, SMILE доступний для практично будь-яких мов і систем. SMILE може бути вбудований в програми, що використовують ймовірнісно-графічні моделі в якості движка для виведення. SMILE випущений у вигляді динамічної бібліотеки (DLL). Також є кілька оболонок (SMILE.NET, SMILEX, jSMILE і ін.). Великою перевагою GeNIe та SMILE є те, що це безкоштовне ПЗ для науково-дослідних і навчальних цілей. Ще одним плюсом є робота з форматами, підтримуваними в інших програмах (наприклад Hugin Expert і Netica).

2.3.5 Hugin-Expert

Робота над HUGIN [40] почалася завдяки проекту Esprit MUNIN, що мав на меті створення експертної системи, яка допоможе лікарям діагностувати захворювання м'язів та нервів. Проект MUNIN діяв у 1984-1989 рр. в Ольборзькому університеті і отримав визнання критиків на Міжнародній конференції зі штучного інтелекту в 1989 р. Після такого успіху проекту технологію вирішили комерціалізувати та створили HUGIN Expert.

У даного програмного забезпечення є безкоштовна версія Hugin Lite (Рисунок 13). Проте вона має обмеження на кількість станів (не більше 50) та об’єму навчальної вибірки (до 500).

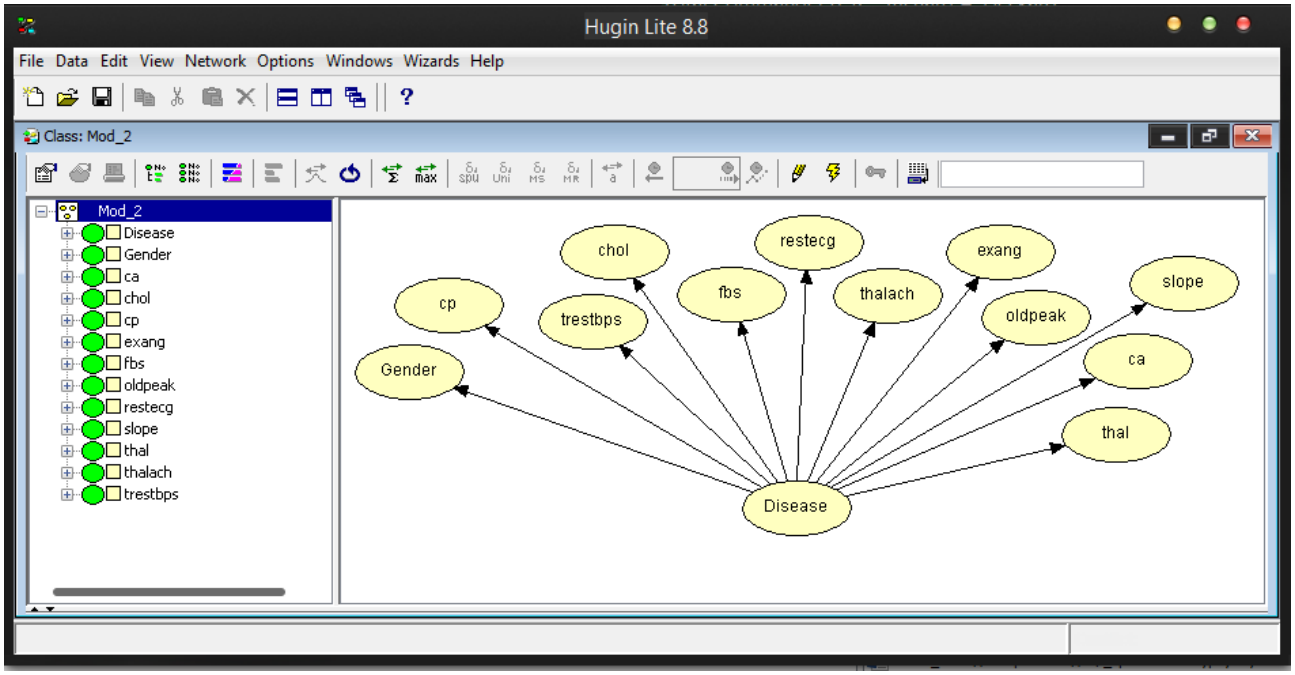

Рисунок 13– Інтерфейс Hugin Lite

2.3.6 SAS Enterprise Miner

SAS Enterprise Miner – спеціалізований інструмент призначений для спрощення процесу аналізу даних, при створенні високоточних інтелектуальних і описових моделей, заснованих на великих обсягах даних [26] (Рисунок 14).

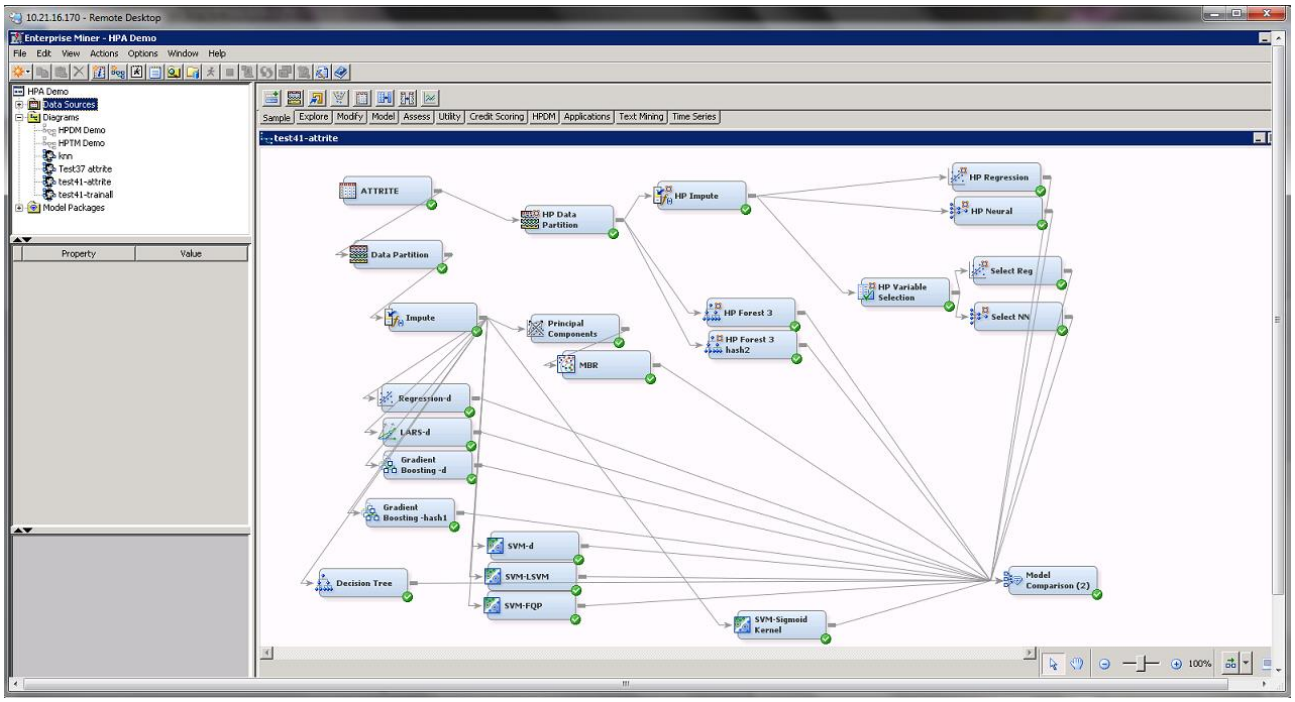

Рисунок 14 – Інтерфейс SAS Enterprise Miner

Серед пропонованих засобів обробки великих обсягів даних є нейронні мережі, кластерний аналіз, дерева рішень, градієнтний бустинг, байєсівські мережі, регресійний аналіз та інші. Швидка робота середовища забезпечується модулем високопродуктивної обробки даних, який працює на основі технологій багатопотоковості та розподіленої обробки даних, використовуючи наявні процесорні можливості та пам’ять комп’ютера. Також є можливість використовувати дану технологію на Hadoop кластерах, таких як Cloudera або Hortonworks, або на виділеному апаратному забезпечені від Teradata, Pivotal та Oracle.

2.4 Висновки до розділу 2

Теорія байєсівських мереж почала активно розвиватися у 1980-х роках. Їх популярність перш за все пов’язана з тим, що БМ враховують причинно-наслідкову природу процесів, можуть будуватися як на основі експертних

оцінок, так і за вибіркою даних, забезпечують наочне графічне представлення досліджуваного об’єкта. Для побудови та оцінювання структури байєсівських мереж за даними існує дві основні групи алгоритмів: на основі скорингових функцій та тестів на умовну незалежність. Таким чином, ці методи враховують відмінності в умовах побудови БМ, наприклад, розмір вибірки даних, тип структури, неповноту спостережень тощо. На сьогоднішній день мережі Байєса застосовуються у різноманітних сферах, зокрема й банківській справі, інженерії, логістиці тощо. Широкого використання вони набули й у медицині, де вирішують задачі діагностики, дозування та взаємодії лікарських засобів, ефективної економічної організації роботи медичних закладів та ін. Найчастіше у даній сфері байєсівські мережі застосовуються у ролі складової СППЛР, які можуть ґрунтуватися як на знаннях, так і на даних про прецеденти.

Завдяки такій популярності БМ на сьогоднішній день розроблено багато програмних продуктів для їх реалізації. Хоча вони є комерційними, у багатьох постачальників існують і безкоштовні версії з обмеженим функціоналом та суттєві знижки для акредитованих вищих навчальних закладів.

РОЗДІЛ 3
РОЗРОБКА ДІАГНОСТИЧНИХ СИСТЕМ ТА РЕЗУЛЬТАТИ ОБЧИСЛЮВАЛЬНИХ ЕКСПЕРИМЕНТІВ

У цьому розділі описується розробка двох медичних діагностичних систем на основі байєсівських мереж та наводяться результати обчислювальних експериментів.

3.1 Вимоги до обладнання та інструменти для роботи з даними

Для обробки вхідних даних було обрано мову програмування Python 3, оскільки вона має зручні вбудовані бібліотеки для аналізу даних, які й було використано. Наведемо їх у таблиці 2.

Таблиця 2 – Використані бібліотеки Python 3

Назва бібліотеки	Версія	Призначення бібліотеки
pandas	0.25.3	Робота з файлами та аналіз даних
seaborn	0.9.0	Візуалізація статистичних даних
matplotlib	3.1.3	Графічне представлення даних
pandas_profiling	2.5.0	Відомості про дані

Для врахування причинно-наслідкового зв'язку між захворюванням та симптомами, які воно спричиняє було вирішено використати байєсівські мережі.

Для побудови БМ обрано середовище Hugin Lite 8.8, оскільки воно є безкоштовним, містить необхідні функції (наприклад, визначення структури та навчання за даними), хоча й має обмеження на кількість вузлів (50) та записів навчальних даних (500), має зручний та зрозумілий інтерфейс, не потребує значного обсягу пам’яті для встановлення.

Для роботи із створеними системами необхідне таке середовище:

- операційна система: Windows, Linux, MacOs;
- програмне середовище Hugin Lite 8.8;
- мережа у вигляді файлу з розширенням .oobn.

3.2 Опис архітектури СППР і функціональної схеми

З технічної точки зору функціонування системи поділяється на такі блоки:

- отримання даних;
- обробка даних;
- прийняття рішення;
- представлення рішення користувачу.

Загальне функціонування системи (Рисунок 15) може бути описане таким чином:

- користувач вводить (обирає) дані (симптоми);
- у блоці інтерпретації та аналізу даних для введених даних з бази даних обираються відповідні функції правдоподібності;
- далі отримані дані подаються на вхід моделі для обчислення апостеріорних ймовірностей та прийняття рішення;

– отримане рішення виводиться користувачу.

Усі описані процеси відбуваються у середовищі Hugin Lite 8.8 на основі завантаженої побудованої та навченої БМ.

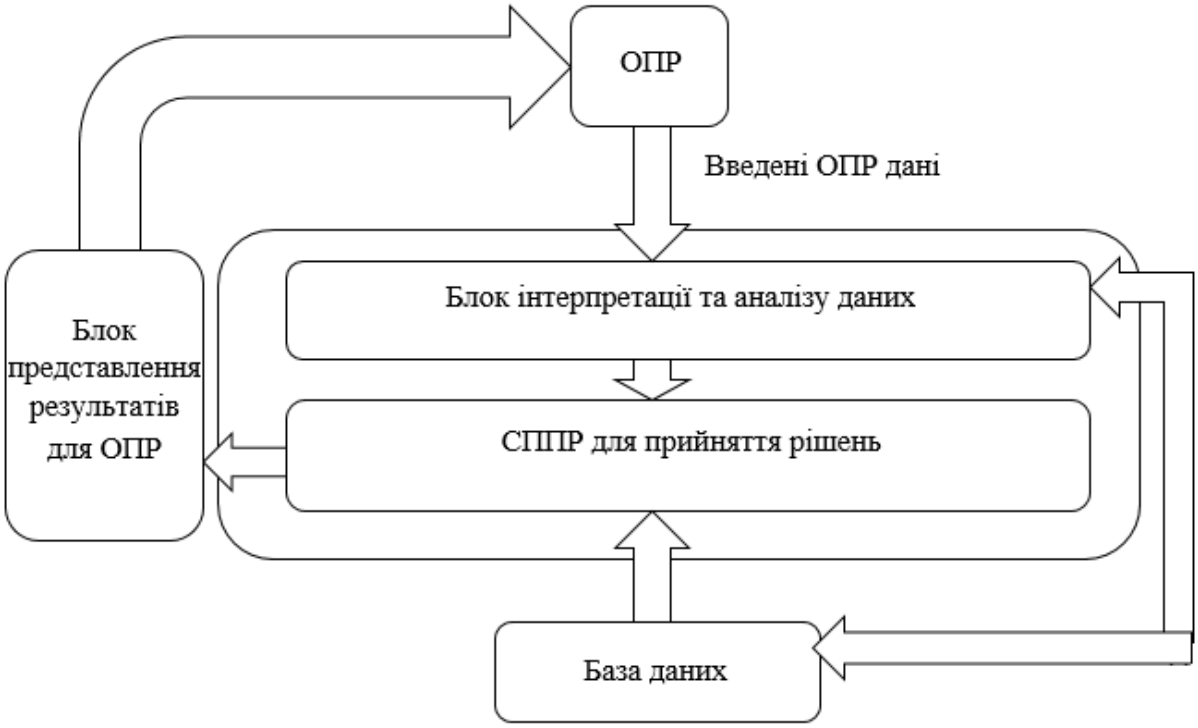

Рисунок 15- Функціональна блок-схема архітектури побудованої СППР

3.3 Система для діагностики наявності хвороб серця

Щороку у світі від серцево-судинних захворювань помирає майже 18 мільйонів осіб. В Україні вони є причиною смерті у двох третинах випадків.

На наявність хвороб серця можуть вказувати багато симптомів (наприклад, аритмія, біль у грудях, підвищений або понижений артеріальний тиск). Проте деякі з них можуть бути тимчасовими і виникати навіть у здорових людей. Тому розроблена система буде допомагати лікарю

діагностувати наявність або відсутність хвороб серця у пацієнта з заданими симптомами.

3.3.1 Підготовка даних

Для визначення структури, навчання мережі та перевірки її роботи був обраний набір даних медичного центру в Клівленді [https://www.kaggle.com/ronitf/heart-disease-uci], що містить триста три записи тринадцяти характеристик пацієнтів (6 числових та 7 категоріальних) та інформацію про наявність або відсутність хвороби серця, що є цільовою змінною. Пропущених даних немає.

Перш за все перевіримо обчислену для вибраних змінних матрицю коефіцієнтів кореляції Пірсона (Рисунок 16).

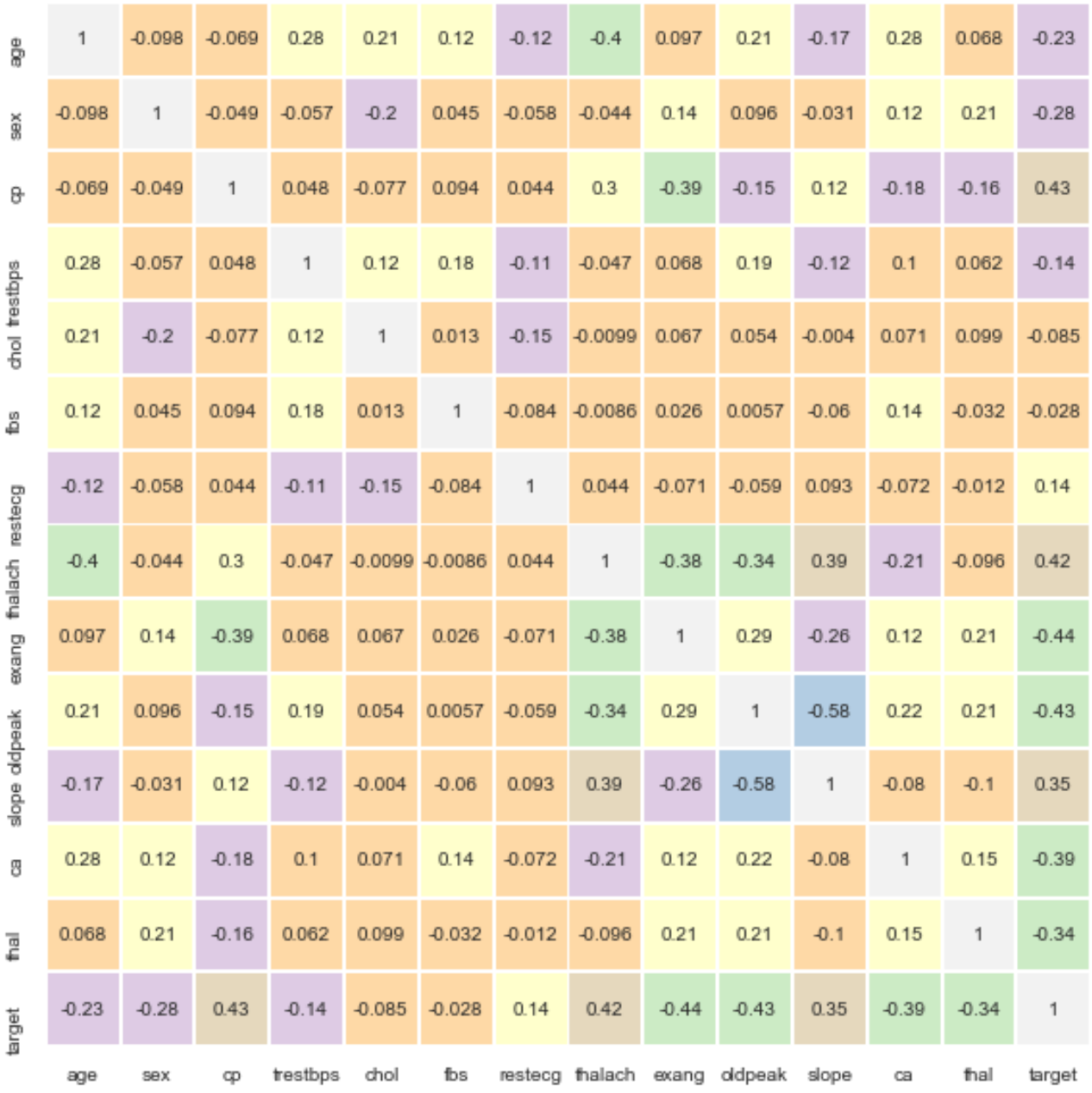

Рисунок 16 - Матриця коефіцієнтів кореляції Пірсона

Таким чином можемо бачити, що коефіцієнт кореляції перевищує значення 0.5 лише між змінними slope та oldpeak. За результатами визначення коефіцієнтів важливості за методом Extra Trees Classifier, залишаємо slope.

Перед застосуванням методів визначення структури мережі та її навчання необхідно перетворити неперервні змінні у категоріальні. Для цього застосовуємо метод дискретизації, що передбачає поділ на інтервали

однакової ширини. Зважаючи на розмір вибірки, бачимо, що необхідна кількість інтервалів лежить у межах від 5 до 10. Перевіривши результати дискретизації для кожного цілого числа із заданого проміжку, обираємо поділ на 5 інтервалів однакової ширини, оскільки при інших значеннях виникає проблема наявності порожніх проміжків.

Після виконаної дискретизації дані набувають вигляду, поданого у таблиці 3.

Таблиця 3 – Опис даних

Назва	Тип	Значення
age_cat	Категоріальна змінна (5)	Вік пацієнта: 0 – 29 – 38; 1 – 39 – 48; 2 – 49 – 58; 3 – 59 – 68; 4 – 69 – 78.
sex	Категоріальна змінна (2)	Стать пацієнта: 0 – жіноча; 1 – чоловіча.
cp	Категоріальна змінна (4)	Біль у грудях: 0 – безсимптомний; 1 – атиповий стенокардичний; 2 – не стенокардичний; 3 – типовий стенокардичний.
trestbps_cat	Категоріальна змінна (5)	Систолічний артеріальний тиск у спокої, (мм. рт. ст. при надходженні до лікарні): 0 – 94 – 115.2; 1 – 115.2 - 136.4; 2 – 136.4 - 157.6; 3 – 157.6 - 178.8; 4 – 178.8 - 200.

Продовження таблиці 3

Назва	Тип	Значення
chol_cat	Категоріальна змінна (5)	Холесторальна сироватка, мг/дл: 0 – 126 - 213.6; 1 – 213.6 - 301.2; 2 – 301.2 - 388.8; 3 – 388.8 - 476.4; 4 – 476.4 – 564.
fbs	Категоріальна змінна (2)	Рівень цукру в крові натще > 120 мг/дл: 0 – ні; 1 – так.
restecg	Категоріальна змінна (3)	Результати електрокардіограми у спокої: 0 – виявлення ймовірної або точної гіпертрофії лівого шлуночка за критерієм Естеса; 1 – норма; 2 – є аномалія хвилі ST-T.
thalach_cat	Категоріальна змінна (5)	Максимальний досягнутий пульс, уд./хв: 0 – 71 - 97.2; 1 – 97.2 - 123.4; 2 – 123.4 - 149.6; 3 – 149.6 - 175.8; 4 – 175.8 - 202.
exang	Категоріальна змінна (2)	Стенокардія, викликана фізичними вправами: 0 – ні; 1 – так.
slope	Категоріальна змінна (3)	Нахил лівого сегмента ST, викликаний вправами: 0 – спадає; 1 - плоский; 2 – зростає.

Кінець таблиці 3

Назва	Тип	Значення
ca_cat	Категоріальна змінна (5)	Кількість основних судин, забарвлених флуороскопією: 0, 1, 2, 3, 4.
thal	Категоріальна змінна (4)	Талій: 0, 1, 2, 3.
target	Категоріальна змінна (2)	Цільова змінна: 0 – хвороби серця немає; 1 – хвороба серця є.

Розглянемо деякі відомі зв’язки між змінними. Очевидно, що наявність хвороби серця впливає на прояв симптомів. Вік та стать пацієнта є одними з факторів, що безпосередньо пов’язані з ризиком розвитку проблем із серцем. Вони також впливають на тиск, рівень холесторальної сироватки та кількість основних судин, що забарвлюються при проведенні флуороскопії.

3.3.2 Побудова, навчання мережі та аналіз її ефективності

Визначення структури байєсівської мережі здійснювалося на основі вбудованих методів програмного середовища Hugin Lite за допомогою Structural Learning Wizard, що працюють на основі набору даних. Результати проведеної роботи наведені у таблиці 4.

Таблиця 4 – Вибір структури мережі

Назва методу	BIC	AIC
PC	-4927.66	-3793.12
NPC	-5026.18	-3817.36
Greedy search-and-score algorithm	-3620.61	-3516.63

Кінець таблиці 4

Назва методу	BIC	AIC
Chow-Liu Tree	-3713.21	-3486.67
Rebane-Pearl Polytree	-3857.62	-3527.1
Tree Augmented Naive Bayes	-4077.03	-3529.25

Отже, можемо бачити, що за значеннями BIC та AIC найкращою є мережа, побудована за методом NPC (Рисунок 17).

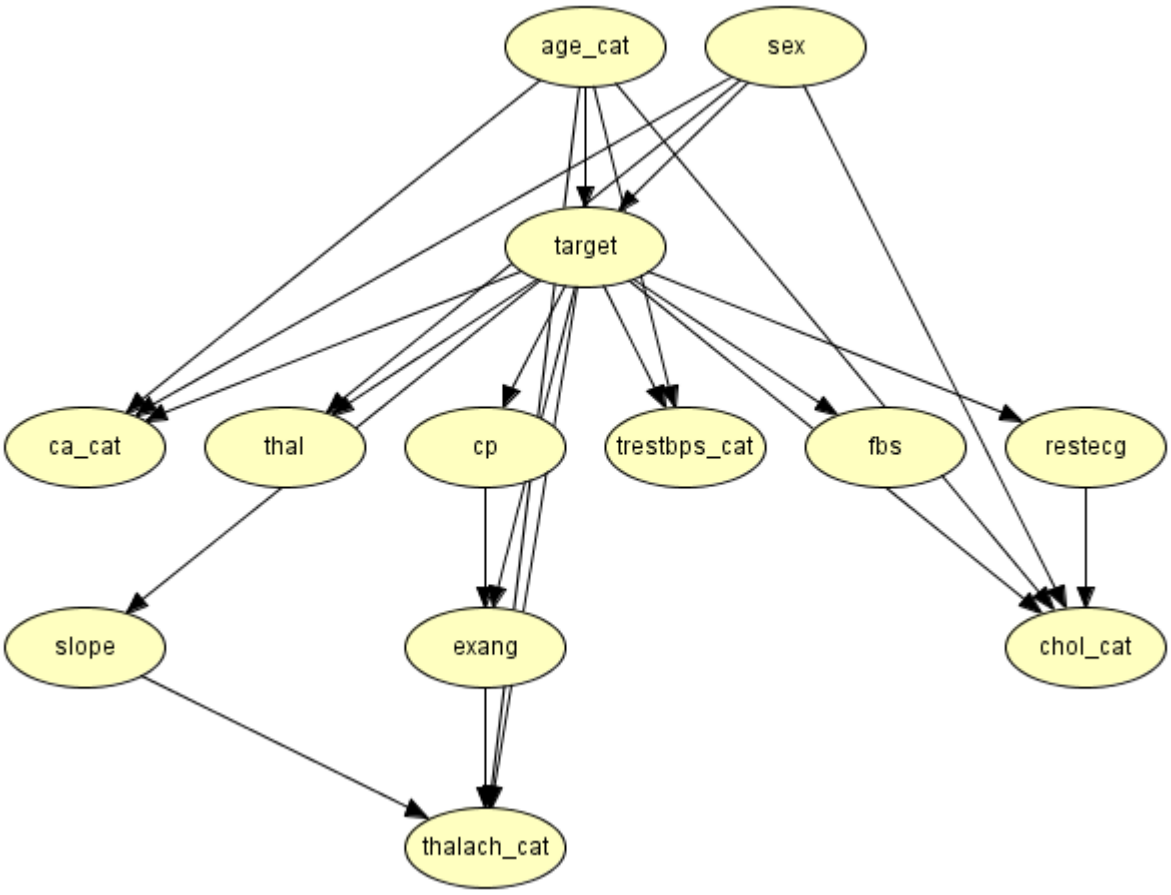

Рисунок 17 – БМ, побудована за методом NPC

Перед початком навчання поділимо дані на навчальну та тестову вибірки у співвідношенні 8:2. Таким чином навчальна вибірка складатиметься зі 242 записів, а тестова – з 61.

Для навчання отриманої мережі скористаємося вбудованою функцією Hugin Lite за допомогою Learning Wizard. У результаті отримуємо заповнені таблиці умовних та безумовних ймовірностей (Додаток Б).

Наступним етапом є перевірка точності роботи отриманої БМ. Для цього також скористаємося вбудованим функціоналом обраного програмного середовища за допомогою Analysis Wizard. Проведемо перевірку на основі отриманої раніше тестової вибірки (Рисунок 18).

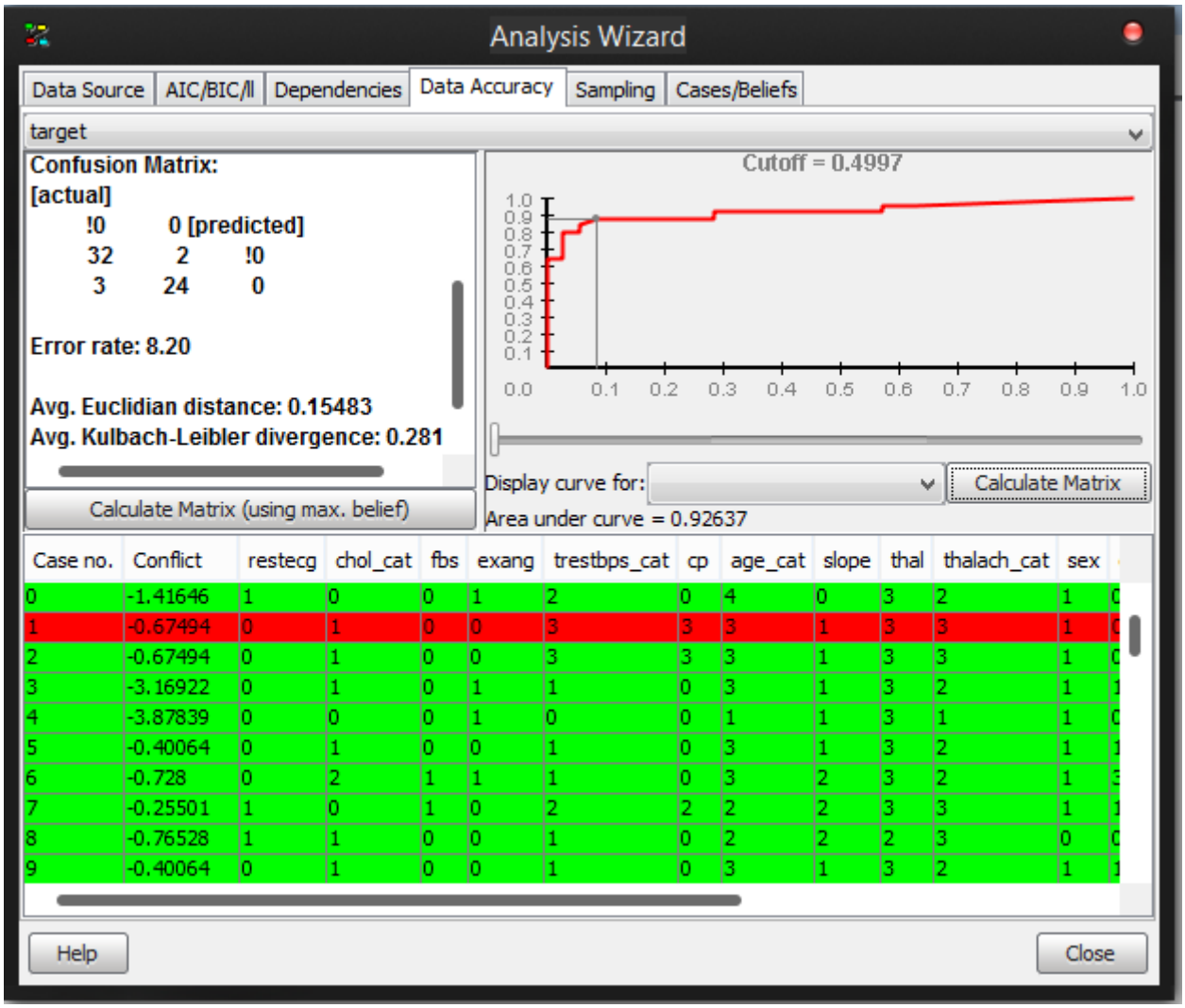

Рисунок 18 – Результати перевірки роботи навченої БМ

Для визначення точності роботи мережі скористаємося показником AUC (area under ROC curve, площа під ROC-кривою (receiver operating characteristic, робоча характеристика приймача або крива похибок)).

Можемо бачити, що точність роботи мережі складає 92.6%, що, безперечно, є хорошим результатом. Неправильно визначених позитивних значень 4.92%, а негативних – 3%.

3.3.3 Приклади роботи системи

Розглянемо деякі приклади роботи побудованої байєсівської мережі.

Приклад 1

Маємо такі дані про пацієнта:

- вік – 49 – 58 років;
- стать – чоловіча;
- кількість основних судин, забарвлених флуороскопією – 0;
- біль у грудях - атиповий стенокардичний;
- систолічний артеріальний тиск у спокої, (мм. рт. ст. при надходженні до лікарні) - 136.4 - 157.6;
- максимальний досягнутий пульс, уд./хв - 175.8 – 202.

Вводимо відомі дані і отримуємо результат: ймовірність наявності хвороби серця становить 92.4% (Рисунок 19).

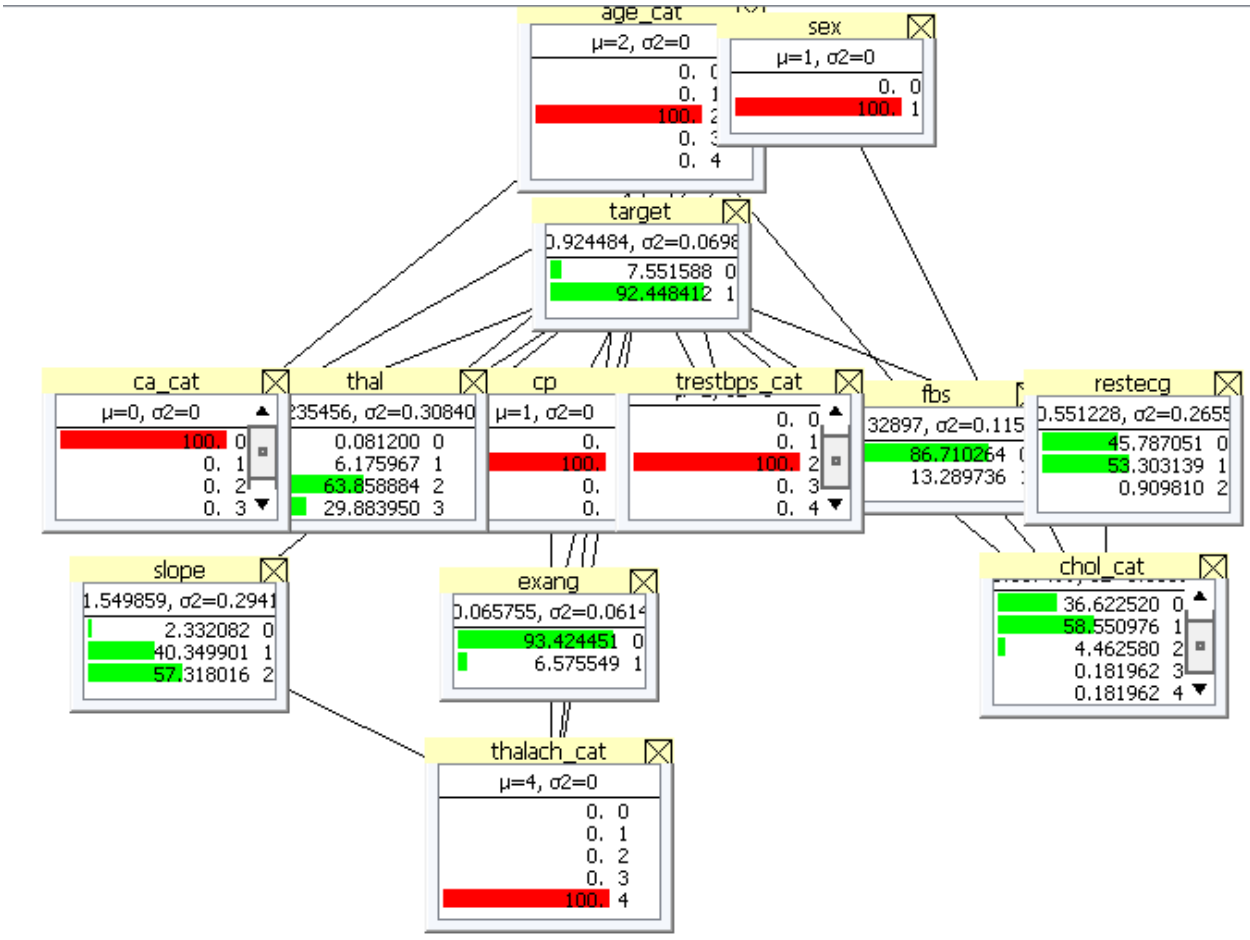

Рисунок 19 – Робота мережі приклад 1

Приклад 2

Маємо такі дані про пацієнта:

- вік – 49 – 58 років;
- стать – чоловіча;
- кількість основних судин, забарвлених флуороскопією – 3;
- біль у грудях - безсимптомний;
- систолічний артеріальний тиск у спокої, (мм. рт. ст. при надходженні до лікарні) - 94 – 115.2;
- максимальний досягнутий пульс, уд./хв - 71 - 97.2.

Вводимо відомі дані і отримуємо результат: ймовірність відсутності хвороби серця становить 98.9% (Рисунок 20).

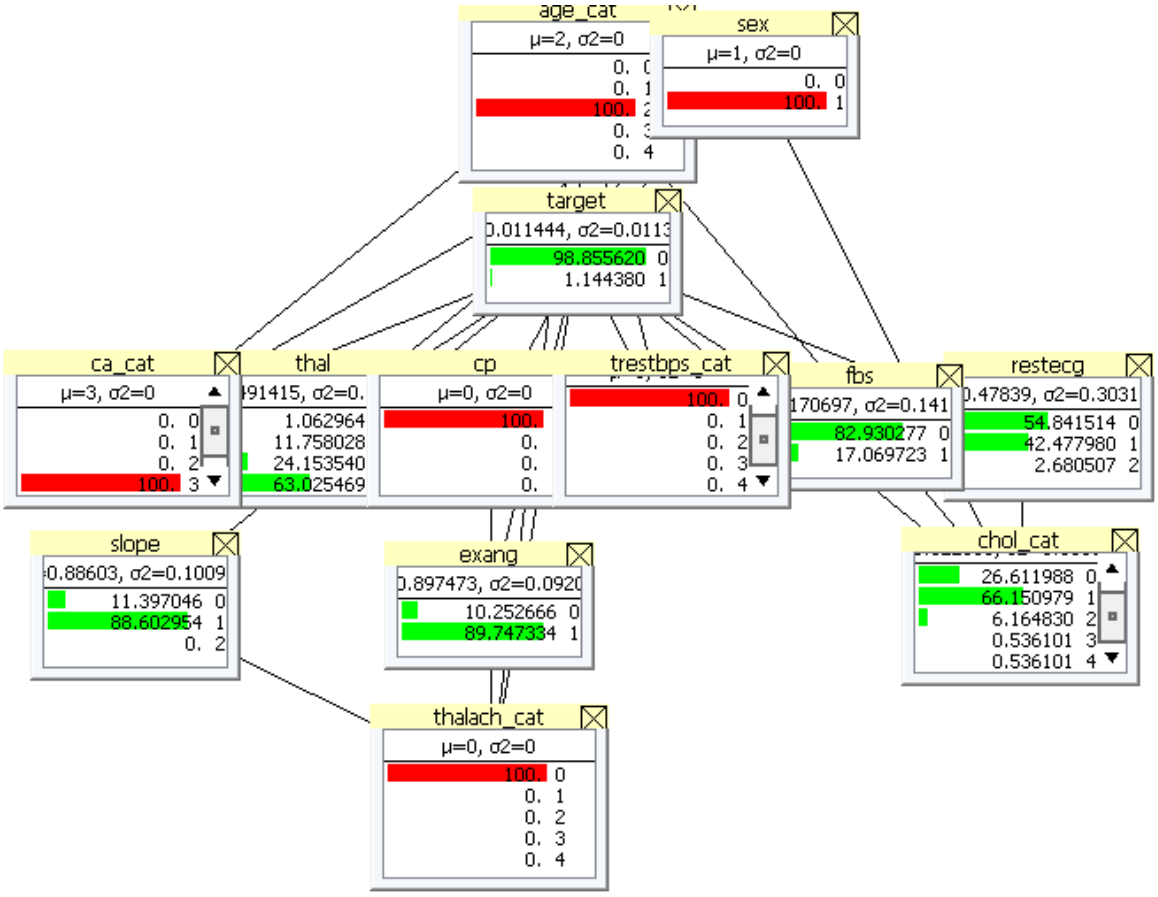

Рисунок 20 - Робота мережі приклад 2

Приклад 3

Маємо такі дані про пацієнта:

- вік – 49 – 58 років;
- стать – чоловіча;
- кількість основних судин, забарвлених флуороскопією – 3;
- біль у грудях - атиповий стенокардичний;
- систолічний артеріальний тиск у спокої, (мм. рт. ст. при надходженні до лікарні) - 136.4 - 157.6;
- максимальний досягнутий пульс, уд./хв - 175.8 - 202.

Вводимо відомі дані і отримуємо результат: ймовірність відсутності хвороби серця становить 46.2%, а наявності 53.8% (Рисунок 21). Отже, результат потребує уточнення, що можна зробити, наприклад, за рахунок додавання відомостей про інші наявні симптоми.

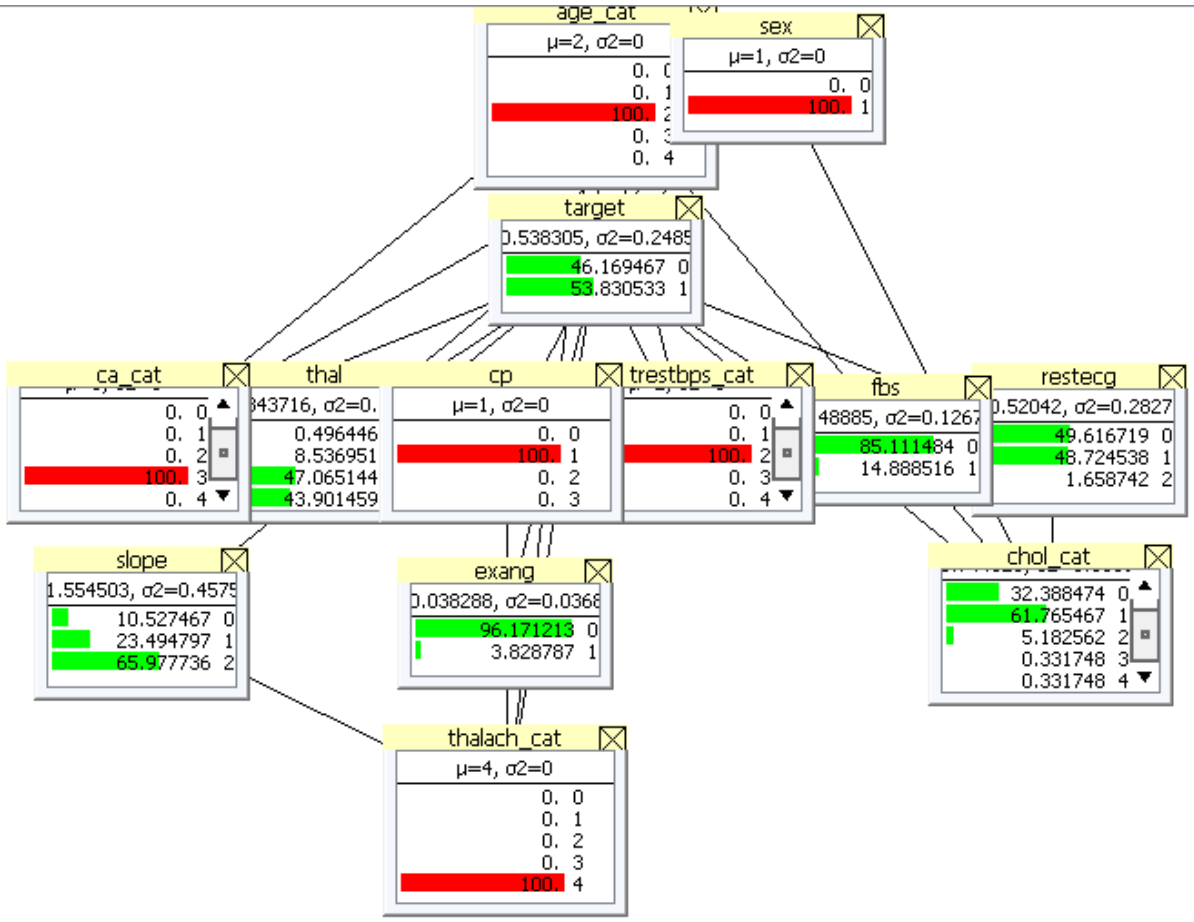

Рисунок 21 - Робота мережі приклад 3

Отримана система може бути використана для попередньої діагностики наявності хвороб серця у пацієнта.

3.4 Система для діагностики наявності COVID-19

Починаючи з грудня 2019 року людство зустрілося з проблемою пандемії вірусу SARS-CoV-2. Через те, що про його існування ніхто раніше не знав, визначення способів поширення, симптомів та пошук ефективного лікування зайняв деякий час. Наразі найбільш дієвими порадами лікарів є дотримання соціальної дистанції, ретельне миття рук, носіння захисних масок та рукавичок, уникання великого скупчення людей. Хоча виконання

таких рекомендацій допомагає зменшити швидкість поширення хвороби, проте воно не може вирішити всіх проблем, однією із яких є діагностика коронавірусу. На сьогодні нам відомі основні симптоми захворювання на COVID-19, якими є сухий кашель, утруднене дихання, в'ялість та підвищена температура. Більшість людей, помічаючи їх у себе, зазвичай звертаються до лікарів. Проте через недостатню кількість тестів та тривалий час їх обробки, а в деяких закладах можливо й через халатність чи некомпетентність персоналу, або й узагалі його відсутність з ряду певних причин, багато випадків захворювання залишаються не зареєстрованими. Таким чином, людина, якій з якоїсь причини не провели тест, або яка чекає на його результати, але вже є хворою, може заразити за цей час ще багато людей. Саме тому проблема діагностики є дуже важливою. Дана система призначена для допомоги лікарю у виявлені захворювання на COVID-19 за вказаними даними про пацієнта.

3.4.1 Підготовка даних

Для проведення дослідження був обраний відкритий набір даних з платформи Kaggle [https://www.kaggle.com/hemanthhari/symptoms-and-covid-presence], що, як зазначає автор, був створений на основі звіту Всесвітньої організації охорони здоров'я за червень 2020 року. Він містить 20 змінних типу boolean, що відображають інформацію про пацієнта, включаючи різні симптоми, а також дані про присутність або відсутність у нього Covid-19. Набір має 5435 записів, проте при дослідженні виявилося, що унікальними є лише 466. Пропущених даних немає. Дві змінні довелося видалити, оскільки всі записи у них містили лише одне однакове значення. Для проведення подальшого дослідження значення "Yes" було замінено 1, а "No" – 0 для усіх змінних.

Розглянемо матрицю коефіцієнтів кореляції Пірсона для отриманих даних (Рисунок 22).

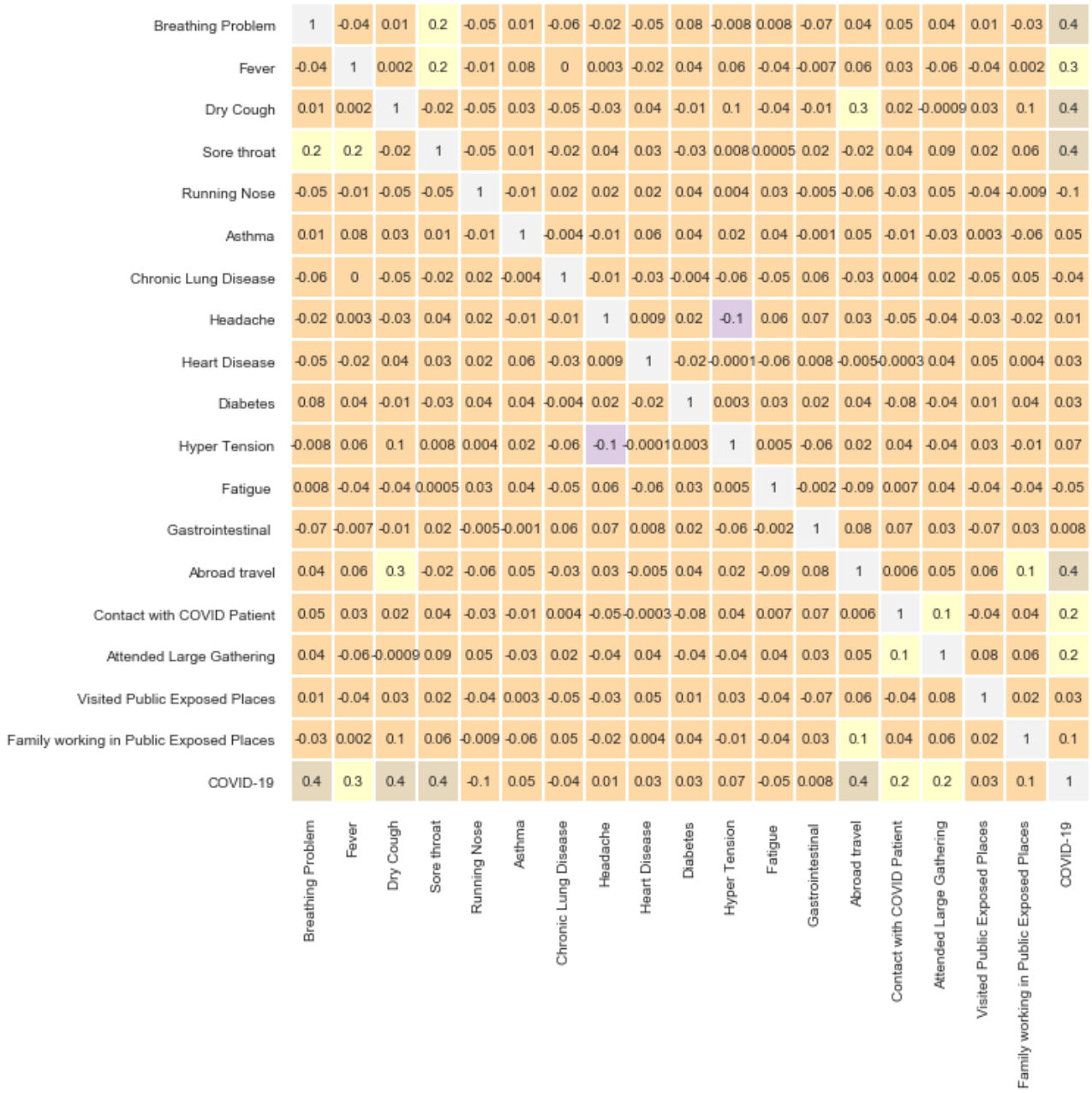

Рисунок 22 - Матриця коефіцієнтів кореляції Пірсона

Можемо бачити, що коефіцієнт кореляції між змінними не перевищує значення 0.4. Таким чином наразі набір даних має вигляд, наведений у таблиці 5. Усі змінні є бінарними, тобто можуть набувати лише одного з двох значень: 0 – ні, 1 – так.

Таблиця 5 – Опис даних

Назва	Значення
Breathing_Problem	Наявні проблеми з диханням
Fever	Жар
Dry_Cough	Сухий кашель
Sore_throat	Біль у горлі
Running_Nose	Нежить
Asthma	Астма
Chronic Lung Disease	Хронічні захворювання легень
Headache	Головний біль
Heart Disease	Хвороби серця
Diabetes	Діабет
Hypertension	Гіпертонія
Fatigue	Втома
Gastrointestinal	Проблеми з шлунково-кишковим трактом
Abroad_travel	Здійснювали закордонні подорожі найближчим часом
Contact_with_COVID_Patient	Контактували з хворим на COVID
Attended_Large_Gathering	Перебували у місцях великого скупчення людей
Visited_Public_Exposed_Places	Відвідували незахищені громадські місця
Family_working_in_Public_Exposed_Places	Члени родини працюють у незахищених громадських місцях
target	Цільова змінна: пацієнт хворий на COVID-19

З таблиці 5 можемо бачити, що у наборі даних присутні змінні, що впливають не на захворюваність COVID-19, а на перебіг хвороби. Оскільки це не є метою дослідження, у даному разі їх доцільно не брати до розгляду, тому остаточний перелік змінних представлений у таблиці 6.

Таблиця 6 – Опис даних для дослідження

Назва	Значення
Breathing_Problem	Наявні проблеми з диханням
Fever	Жар
Dry_Cough	Сухий кашель
Sore_throat	Біль у горлі
Running_Nose	Нежить
Headache	Головний біль
Fatigue	Втома
Gastrointestinal	Проблеми з шлунково-кишковим трактом
Abroad_travel	Здійснювали закордонні подорожі найближчим часом
Contact_with_COVID_Patient	Контактували з хворим на COVID
Attended_Large_Gathering	Перебували у місцях великого скупчення людей
Visited_Public_Exposed_Places	Відвідували незахищені громадські місця
Family_working_in_Public_Exposed_Places	Члени родини працюють у незахищених громадських місцях
target	Цільова змінна: пацієнт хворий на COVID-19

Таким чином остаточний набір даних складається з 14 змінних разом з цільовою і має 466 записів.

Розглянемо деякі відомі зв’язки між змінними. Очевидно, що захворювання на COVID-19 спричиняє наявність відповідних симптомів, що перераховані у таблиці 6. Зважаючи на способи поширення вірусу, можемо сказати, що перебування у місцях великого скупчення людей або незахищених громадських місцях, контактування з хворим на COVID-19 збільшують ризик захворіти. Також знаємо, що біль у горлі зазвичай спричиняє підвищення температури, а нежить та кашель – утруднення дихання.

3.4.2 Побудова, навчання мережі та аналіз її ефективності

Як і для попередньої системи, визначення структури байєсівської мережі здійснювалося на основі вбудованих методів програмного середовища Hugin Lite за допомогою Structural Learning Wizard. Результати проведеної роботи наведені у таблиці 7.

Таблиця 7 – Вибір структури мережі

Назва методу	BIC	AIC
PC	-4712.56	-4584.09
NPC	-4712.56	-4584.09
Greedy search-and-score algorithm	-4626.54	-4556.09
Chow-Liu Tree	-4605.04	-4549.09
Rebane-Pearl Polytree	-4620.51	-4511.09
Tree Augmented Naive Bayes	-4678.77	-4573.09

Отже, можемо бачити, що за значеннями BIC та AIC найкращими є мережі, побудовані за методами NPC та PC, які у даному випадку співпадають (Рисунок 23).

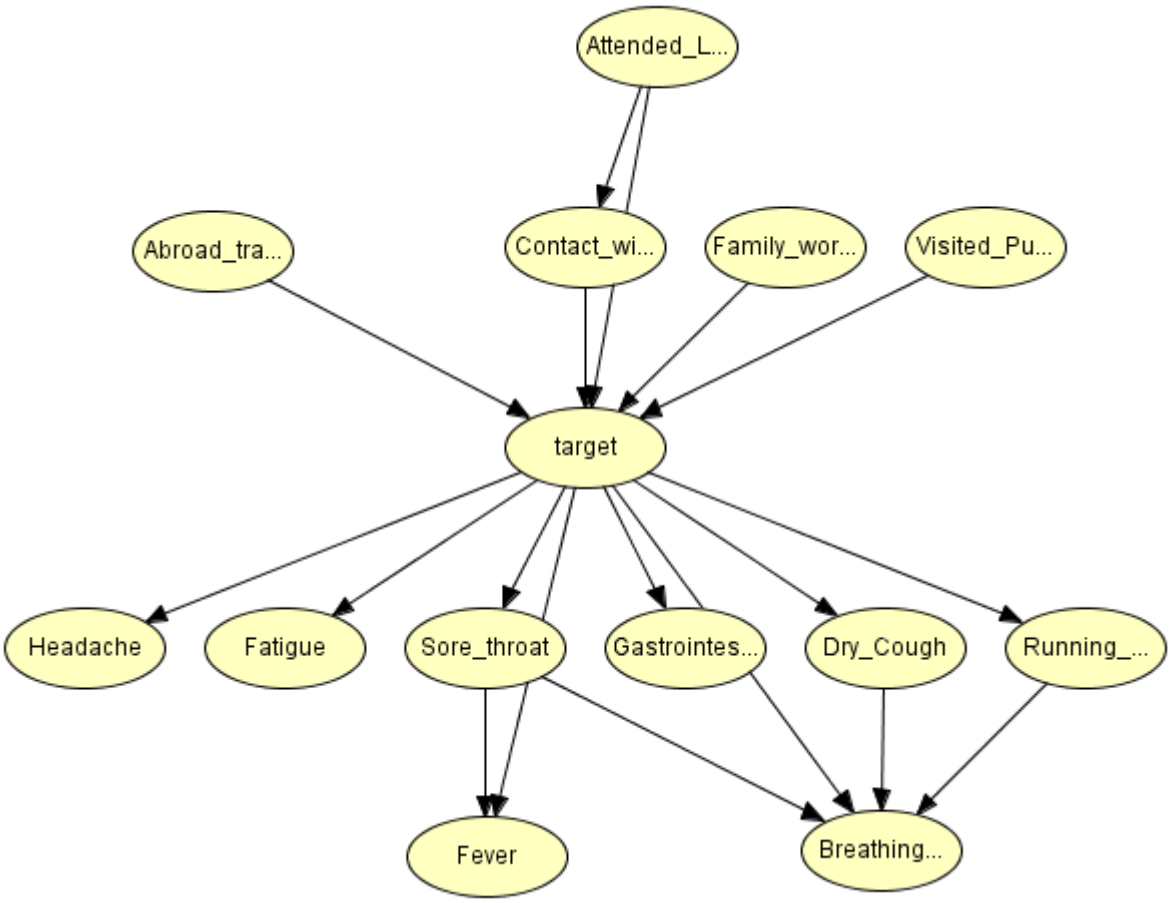

Рисунок 23 – БМ, побудована за методом NPC

Перед початком навчання поділимо дані на навчальну та тестову вибірки у співвідношенні 8:2. Тренувальний набір даних міститиме 372 записи, а тестовий – 94.

Для навчання отриманої мережі скористаємося вбудованою функцією Hugin Lite. У результаті отримуємо заповнені таблиці умовних та безумовних ймовірностей (Додаток В).

Наступним етапом є перевірка точності роботи отриманої БМ. Для цього також скористаємося вбудованим функціоналом обраного програмного

середовища. Проведемо перевірку на основі отриманої раніше тестової вибірки (Рисунок 24).

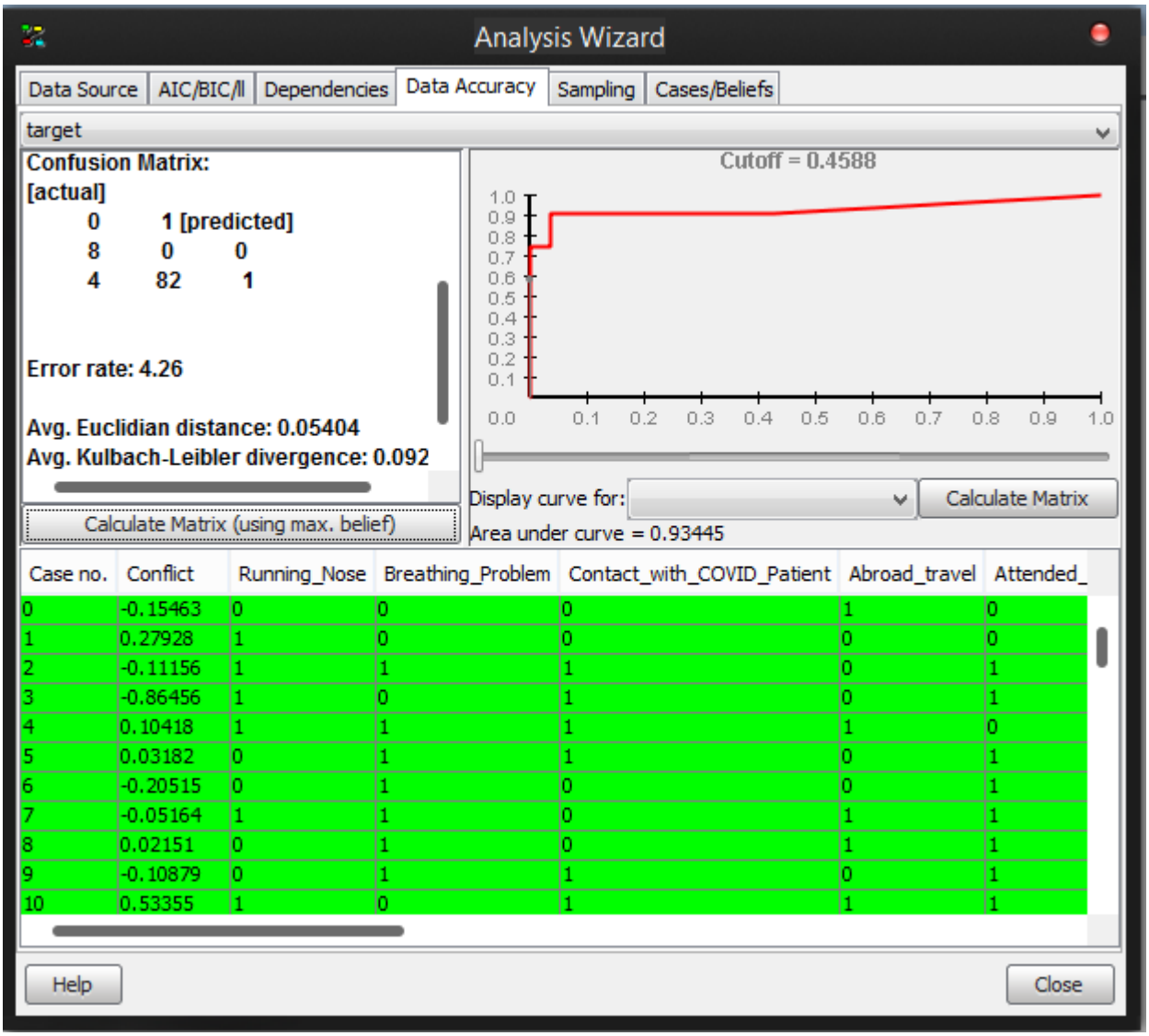

Рисунок 24 – Результати перевірки роботи навченої БМ

Можемо бачити, що точність роботи мережі складає 93.4%, що є хорошим результатом. Неправильно визначених позитивних значень (присутність хвороби) немає, а негативних – 4%.

3.4.3 Приклади роботи системи

Розглянемо деякі приклади роботи побудованої байєсівської мережі.

Приклад 1

Маємо такі дані про пацієнта:

- перебували у місцях великого скупчення людей – так;
- втома – так;
- біль у горлі – так;
- сухий кашель – так;
- жар – так;
- проблеми з диханням – так.

Вводимо відомі дані і отримуємо результат: ймовірність, що пацієнт хворий на COVID-19 становить 98.5% (Рисунок 25).

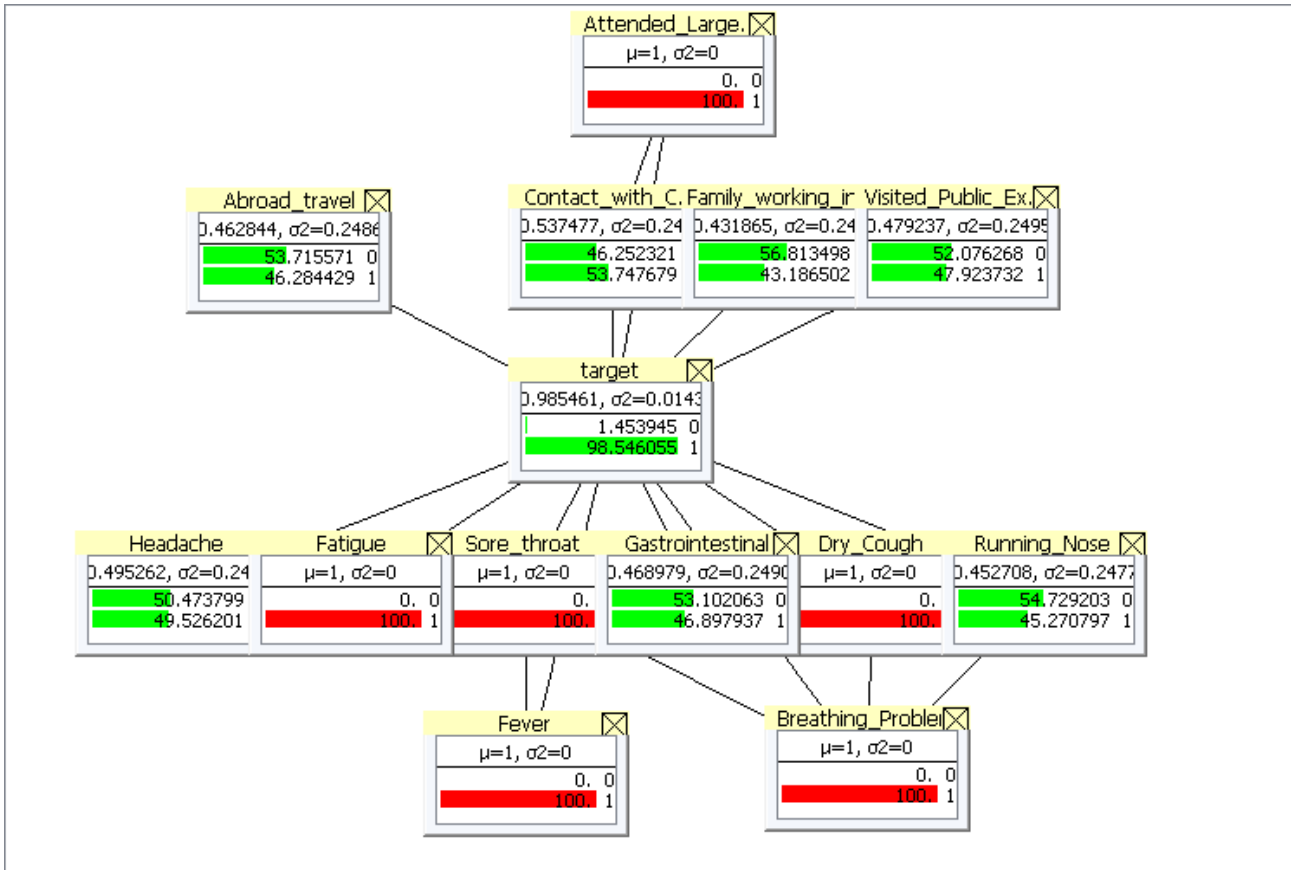

Рисунок 25 – Робота мережі приклад 1

Приклад 2

Маємо такі дані про пацієнта:

– перебували у місцях великого скупчення людей – так;
– головний біль – так;
– втома – так;
– біль у горлі – так;
– сухий кашель – ні;
– жар – так;
– проблеми з диханням – ні.

Вводимо відомі дані і отримуємо результат: ймовірність того, що пацієнт не хворий на COVID-19 становить 83.3% (Рисунок 26).

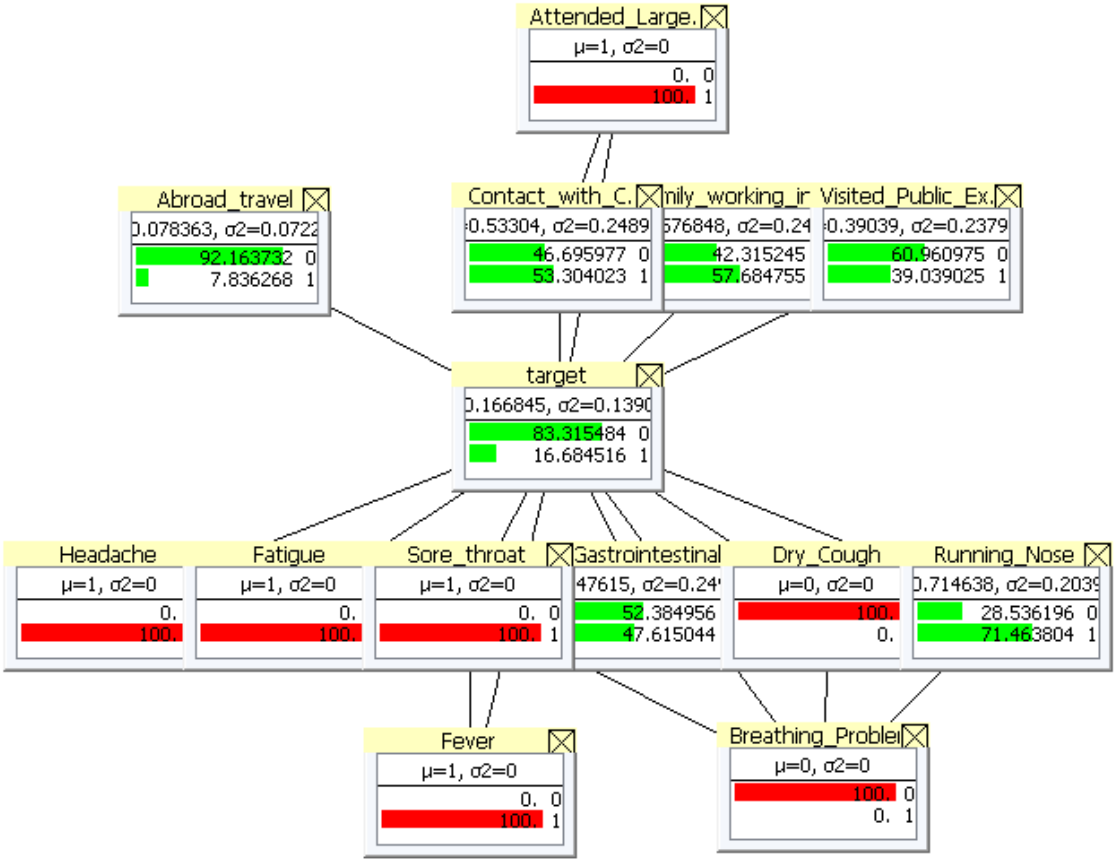

Рисунок 26 - Робота мережі приклад 2

Приклад 3

Маємо такі дані про пацієнта:

- перебували у місцях великого скупчення людей – так;
- головний біль – так;
- втома – так;
- біль у горлі – ні;
- сухий кашель – так;
- нежить – так;
- жар – ні;
- проблеми з диханням – ні.

Вводимо відомі дані і отримуємо результат: ймовірність того, що пацієнт хворий на COVID-19 становить 54.2%, а того, що ні - 45.8% (Рисунок

27). Отже, результат потребує уточнення, що можна зробити, наприклад, за рахунок проведення ПЛР тесту.

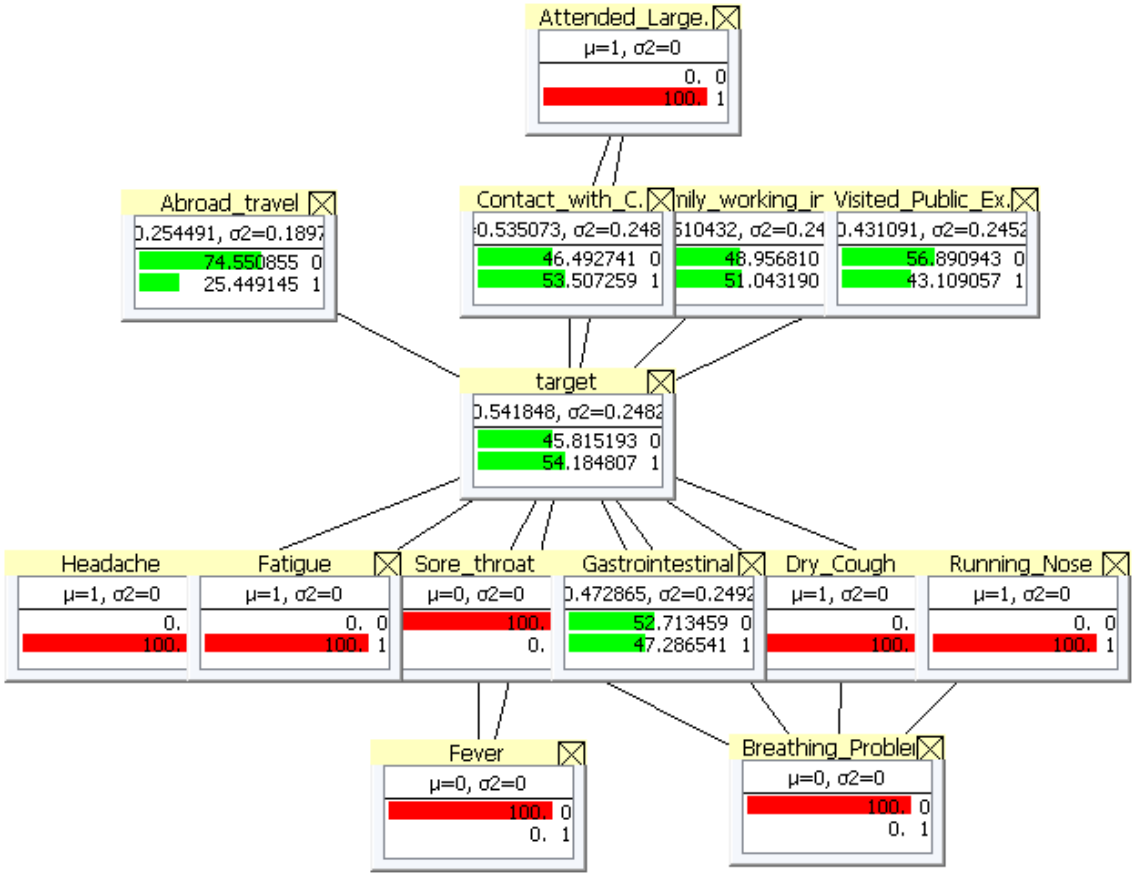

Рисунок 27 - Робота мережі приклад 3

Отримана система може бути використана для попередньої діагностики захворювання пацієнта на COVID-19.

3.5 Висновки до розділу 3

У поданому розділі описано процес створення двох діагностичних систем за допомогою запропонованої системи підтримки прийняття рішень, що включає пошук та обробку даних, вибір мови програмування для роботи з

даними та середовища для побудови і навчання байєсівських мереж, власне побудову мережі, вибір найкращої структури, її навчання та аналіз точності роботи. У розділі також подано по три приклади застосування створених систем для діагностики вибраних захворювань на основі фактичних даних, взятих з відомих баз.

За результатами виконаних обчислювальних експериментів з побудови і застосування ймовірнісно-статистичних моделей у формі мереж Байєса можна стверджувати, що розроблені системи цілком придатні для допомоги лікарям у діагностуванні наявності хвороб серця та COVID-19 відповідно. У майбутніх дослідженнях система буде розширена можливостями діагностування інших видів захворювань.

РОЗДІЛ 4
РОЗРОБКА СТАРТАП-ПРОЕКТУ

У цьому розділі проводиться розробка стартап-проекту на основі створених діагностичних систем.

4.1 Опис ідеї стартап-проекту

Будь-який стартап-проект починається з ідеї. Вона повинна бути чітко сформульована і усвідомлена. Опис ідеї даного стартап-проекту представлений у таблиці 8.

Таблиця 8– Опис ідеї стартап-проекту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Система для діагностики хвороб серця та COVID-19 на основі байєсівських мереж	Попередня діагностика у медичних закладах	Дозволяє швидко обробити наявні дані про пацієнта медпрацівнику з будь-яким рівнем підготовки; інтуїтивно зрозуміле зручне та швидке представлення результатів; додаткове опорне джерело інформації при обранні подальших необхідних дій лікаря.

Для того, щоб визначити, чи варто ідею реалізовувати, її потрібно проаналізувати. Аналіз нашої ідеї проведено у таблиці 9.

Таблиця 9 – Сильні, слабкі та нейтральні характеристики ідеї стартап-проекту

№ п/п	Техніко-економічні характеристики ідеї	(Потенційні) товари/концепції конкурентів			
		Моя система	Protis Assessment	Change Healthcare	Micromedex Clinical Knowledge
1	Ціна	Низька	Висока	Висока	Висока
2	Функціонал	Вузький	Широкий	Широкий	Надширокий

З таблиці 9 можемо бачити, що ціна є, безперечно, сильною характеристикою системи. Зважаючи на область застосувань, функціонал також може стати сильною характеристикою.

4.2 Розробка бізнес-моделі стартапу

Наступним кроком є розробка бізнес-моделі проекту. Вона наведена у таблиці 10.

Таблиця 10 - Бізнес-модель Canvas

№ п/п	Назва	Зміст
1	Споживчі сегменти	Нішовий ринок (медичні заклади).
2	Ключові види діяльності	Система для діагностики хвороб серця та COVID-19.

Продовження таблиці 10

№ п/п	Назва	Зміст
3	Ціннісна пропозиція	Новизна – відсутність подібних пропозицій на ринку України. Зручність – інтуїтивно зрозуміле, зручне та швидке представлення результатів та введення даних. Доступність – завдяки низькій ціні та малому обсягу необхідних комп'ютерних ресурсів. Дозволяє швидко обробити наявні дані про пацієнта медпрацівнику з будь-яким рівнем підготовки. Є додатковим опорним джерелом інформації при обранні подальших необхідних дій лікаря.
4	Канали збуту	Демоверсія системи, реклама.
5	Взаємовідносини з клієнтами	Персональна підтримка на початкових етапах впровадження системи у медичному закладі, далі автоматизоване обслуговування.
6	Потоки надходження доходу	Ліцензія.
7	Ключові ресурси	Матеріальні (комп'ютери, інтернет, програмне середовище для створення БМ). Інтелектуальні (моделі, алгоритми, дані). Людські (бізнес аналітик, розробники, тестувальник, консультант). Фінансові (гроші на розвиток та підтримку системи, ліцензія на комерційне використання програмного середовища для створення БМ).

Кінець таблиці 10

№ п/п	Назва	Зміст
8	Ключові партнери	Відносини виробника з постачальниками для гарантії отримання якісних комплектуючих (розробники програмного забезпечення для побудови БМ).
9	Структура витрат	З переважною увагою до цінності. Витрати: фіксовані (ліцензія на використання ПЗ). Змінні (тех. підтримка, збереження даних, реклама).

4.3 Аналіз ринкових можливостей та розробка маркетингової стратегії стартап-проекту

Проведемо аналіз ринкових можливостей. Для цього розглянемо характеристики потенційних клієнтів нашого стартап-проекту (таблиця 11), визначимо ринкові можливості та загрози (таблиця 12), проведемо аналіз конкуренції в обраній галузі за М. Портером (таблиця 13) та здійснимо обґрунтування факторів конкурентоспроможності (таблиця 14). Підсумуємо все це у SWOT-аналізі стартап-проекту (таблиця 15).

Таблиця 11 - Характеристика потенційних клієнтів стартап-проекту

Потреба, що формує ринок	Цільова аудиторія (цільові сегменти ринку)	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів до товару
Необхідна опорна інформація на основі проаналізованих даних про пацієнта для прийняття лікарських рішень.	Медичні заклади, приватні лікарі.	Ціновий фактор, наявність ліцензії, умови договору з клінікою, функціонал.	Справедливі умови договору з лікарями та медичними закладами. Якісні послуги та надійні моделі. Справедлива ціна. Зрозумілість та зручність використання системи.

Таблиця 12 - Визначення ринкових можливостей і загроз

Параметри оцінки	Можливості	Загрози
1. Конкуренція	Лікарні у малих містах та селах не можуть дозволити собі дороге програмне забезпечення, не потребують широкого функціоналу.	Поява безкоштовної для державних медичних закладів діагностичної системи.

Кінець таблиці 12

Параметри оцінки	Можливості	Загрози
2. Збут	Забезпечення усіх державних лікарень комп'ютерами.	Відсутність у малих лікарнях будь-яких комп’ютерних засобів.
3.Політичні і правові чинники	- .	Заборона використання будь-яких СППР не затверджених урядом.
4.Соціально-культурні чинники	Зростання довіри до СППЛР.	Стереотип про неякісність даних, наданих системою, що не базується на знаннях.
5.Міжнародні чинники	Необхідність швидкої реакції у зв’язку з пандемією.	Всесвітня криза.

Таблиця 13 - Аналіз конкуренції в галузі за М. Портером

	Прямі конкуренти в галузі	Потенційні конкуренти	Постачальники	Клієнти	Товари-замінники
Складові аналізу	MedElement	Бар’єри входження в ринок: затрати, ресурси, попит	Підписання контрактів про співпрацю	Попит	Функціонал

Кінець таблиці 13

	Прямі конкуренти в галузі	Потенційні конкуренти	Постачальники	Клієнти	Товари-замінники
Висновки:	Низька інтенсивність конкурентної боротьби з боку прямих конкурентів.	-є можливості входу в ринок; - немає потенційних конкурентів.	Постачальники не диктують умови роботи на ринку.	Клієнти диктують такі умови роботи на ринку: якість (точність результатів), зручність у використанні, доступність.	Обмеження для роботи на ринку через товари замінники: обмеження у збалансованості ціни та функціоналу.

Таблиця 14 - Обґрунтування факторів конкурентоспроможності

№ п/п	Фактор конкурентоспроможності	Обґрунтування
1	Фактор новизни	Продукт може зацікавити своєю новизною та додатковими можливостями порівняно з конкурентами.
2	Простота у використанні	Продукт максимально user friendly та не перевантажений зайвою інформацією.
3	Вартість	Не завищена, конкурентна ціна.
4	Рівень якості послуг	Точність роботи системи на рівні не менше 90%.

Таблиця 15 - SWOT- аналіз стартап-проекту

	Сильні сторони **1.Відсутність продуктів з подібним функціоналом на ринку України.** **2.Висока якість системи.** **3.Зручність у використанні.** **4.Помірна ціна.**	**Слабкі сторони** **1.Відсутність репутації.** **2.Неможливість використання без комп'ютера з встановленим необхідним ПЗ.** **3.Вузький функціонал.**
Можливості **1.Поява попиту на системи підтримки прийняття рішень у зв'язку з пандемією.** **2.Комп'ютеризація державних медичних закладів.**	З появою попиту на СППЛР наша система, завдяки відсутності подібних пропозицій, високій якості послуг та зручності у використанні займе вигідну позицію на ринку. Комп'ютеризація державних медичних закладів завдяки помірній ціні продукту розширить коло потенційних клієнтів.	Відсутність репутації можна компенсувати за рахунок появи попиту через пандемію, що покриває вже створений функціонал. Комп'ютеризація державних медичних закладів усуне проблему з необхідністю комп'ютера для користування системою.
Загрози **1.Вхід на український ринок закордонних постачальників.** **2. Стереотип про неякісність даних, наданих системою, що не базується на знаннях.**	Помірна ціна буде сильною стороною в порівнянні з іноземними системами. Висока якість системи допоможе збільшити довіру до систем, що не базуються на знаннях.	Збільшення довіри та попиту допоможе швидко набути репутацію. Вузький функціонал обумовлює нижчу ціну.

4.4 Розробка маркетингової програми стартап-проекту

Складові розробки маркетингової програми нашого стартап-проекту наведемо у таблицях 16 – 21.

Таблиця 16 - Визначення базової стратегії розвитку

Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку
Цільовий маркетинг	Якісна, зручна у використанні система за помірною ціною.	Позиціонування за співвідношенням «ціна-якість»

Таблиця 17 - Визначення базової стратегії конкурентної поведінки

Чи є проект «першопрохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки
Так	Шукати нових	Не буде	Стратегія заняття конкурентної ніші

Таблиця 18 - Визначення ключових переваг концепції потенційного товару

Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити)
Необхідна опорна інформація на основі проаналізованих даних про пацієнта для прийняття лікарських рішень.	Швидка, надійна, зрозуміла та зручна у використанні діагностична система.	Низька ціна.

Таблиця 19 - Опис трьох рівнів моделі товару

Рівні товару	Сутність та складові
І. Товар за задумом	Швидкий аналіз даних про пацієнта та визначення ймовірності наявності у нього певної хвороби.
ІІ. Товар у реальному виконанні	Діагностична система, що за введеними симптомами визначає ймовірність явності хвороби у пацієнта.
	Малий розмір ПЗ.
	Інтуїтивно зрозумілий інтерфейс, зручне введення інформації.
ІІІ. Товар із підкріпленням	До продажу: наявність комп’ютера зі встановленим необхідним ПЗ.
	Після продажу: ліцензія на використання, технічна підтримка.

Таблиця 20 - Формування системи збуту

Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
Клієнти не користуються системами підтримки прийняття рішень.	Встановлення контактів зі споживачами та їх підтримка.	Канал нульового рівня.	Реклама діагностичної системи, демоверсія.

Таблиця 21 - Концепція маркетингових комунікацій

Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуються цільові клієнти	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення	Концепція рекламного звернення
Клієнти дізнаються про систему через рекламу, на спеціальних з'їздах та форумах.	Інтернет, спільноти працівників медичних закладів.	Контент-маркетинг, публікації у спеціалізованій літературі.	Інформувати про наявність, можливості та переваги діагностичної системи підтримки прийняття лікарських рішень.	"Надійна підтримка: якісний аналіз симптомів за лічені секунди!"

4.5 Висновки до розділу 4

У цьому розділі проаналізовано можливість впровадження створених діагностичних систем у вигляді стартап-проекту. Для цього здійснено аналіз ідеї, розроблено бізнес-модель, проаналізовано ринкові можливості та створено маркетингову програму стартапу.

У результаті проведеної роботи можемо сказати, що на ринку України наш проект з заданим функціоналом буде першопрохідцем, проте це не стане перешкодою для впровадження, оскільки необхідність у таких системах зростає. Програмні характеристики створеного продукту, інтерфейс та ціна будуть орієнтуватися на українського споживача. Таким чином, навіть при входженні на ринок іноземних постачальників, продукт залишиться конкурентоспроможним. Таким чином подальша імплементація проекту є доцільною.

ВИСНОВКИ

У ході виконання магістерської дисертації в результаті аналізу актуальності проблеми підвищення якості оцінювання стану пацієнта було виявлено, що смертність спричинена неправильністю поставленого діагнозу або призначеного лікування є високою навіть у розвинених країнах світу. В Україні це також є проблемою, тому її вирішення є дуже актуальним.

Проведений аналіз літератури та існуючих результатів показав, що для допомоги лікарям ще з 1970-х років створюються системи підтримки прийняття рішень. Вони будуються на основі систем правил, нейронних мереж та інших методів.

Для побудови власної діагностичної системи було обрано два відкритих набори даних, що містили інформацію про пацієнта та висновки про наявність хвороб серця або COVID-19 відповідно. Дані було проаналізовано на унікальність записів, повноту, залежності між змінними. Для їх подальшого використання неперервні змінні було дискретизовано.

На основі оброблених даних у програмному середовищі Hugin Lite 8.8 було побудовано відповідні байєсівські мережі. Для вибору кращої структури мережі використовувалися інформаційні критерії Байєса та Акайке.

Для тестування роботи отриманих мереж була використана частина оброблених даних, що не входила до навчального набору. Точність визначення стану пацієнта оцінювалася за значенням показника площі під кривою похибок та матрицею помилок. Обидві мережі показали хороший результат на рівні 90%. Таким чином створені системи можна використовувати для попередньої діагностики у медичних закладах.

В подальшій роботі необхідно звернути увагу на такі питання:

- збільшення кількості пропонованих для діагностики захворювань;
- додавання функціоналу збереження отриманих результатів;
- додавання функціоналу персоналізації результатів;
- покращення точності отриманих мереж за рахунок нових навчальних даних.

ПЕРЕЛІК ПОСИЛАНЬ

1. World Health Organisation. The Conceptual Framework for the International Classification for Patient Safety. Geneva: World Health Organization, 2009. 154 p.
2. Prevalence, severity, and nature of preventable patient harm across medical care settings: systematic review and meta-analysis / M. Panagioti et. al. *BMJ*. 2019.
   URL: https://www.bmj.com/content/366/bmj.l4185 (accessed: 15.09.2020).
3. Статистика Врачебных Ошибок. URL: https://vawilon.ru/statistika-vrachebnyh-oshibok/ (accessed: 15.09.2020).
4. В Україні хочуть ввести обов'язкове страхування лікарської відповідальності. Що це означає? URL: https://nv.ua/biz/experts/oshibki-ukrainskih-vrachey-pomozhet-li-obyazatelnoe-strahovanie-mnenie-eksperta-50067523.html (accessed: 15.09.2020).
5. Greenes R. A. Clinical decision support: the road ahead. Boston: Elsevier Academic Press, 2007. 581 p.
6. Литвин А. А., Литвин В. А. Системы поддержки принятия решений в хирургии. *Новости хирургии*. 2014. Т. 22, № 1. С. 96–100.
7. Система поддержки принятия врачебных решений. URL: https://ru.wikipedia.org/wiki/Система_поддержки_принятия_врачебных_решений#cite_note-Berner-7 (accessed: 15.09.2020).
8. Доан Д.Х., Крошилин А. В., Крошилина С. В. Обзор подходов к проблеме принятия решений в медицинских информационных системах в условиях неопределенности. *Фундаментальные исследования*. 2015. № 12. С. 26–30.

9. Раводин Р. А. Интеллектуальная система поддержки принятия врачебных решений в дерматовенерологии. *Проблемы медицинской микологии*. 2014. Т. 16, № 3. С. 59–65.
10. Зарипова Г. Р., Богданова Ю. А., Катаев В. А., Ханов В. О. Современные модели экспертных систем поддержки принятия врачебных решений в прогнозировании операционного риска в хирургической практике. *Таврический медико-биологический вестник*. 2016. Т. 19, № 4. С. 140–145.
11. Купеева И. А., Разнатовский К. И., Раводин Р. А. Разработка интеллектуальной системы поддержки принятия врачебных решений в дерматовенерологии. *Проблемы медицинской микологии*. 2015. Т. 17, № 3. С. 27–31.
12. Атьков О. Ю., Кудряшов Ю. Ю., Прохоров А. А., Касимов О. В. Система поддержки принятия врачебных решений. *Врач и информационные технологии*. 2013. № 6. С. 67–75.
13. Гаврилов Э. Л., Хоманов К. Э., Короткова А. В., Аслибекян Н. О., Шевченко Е. А. Актуальные направления развития справочно-информационных он-лайн приложений для врачей. *Вестник Национального медико-хирургического центра им. Н. И. Пирогова*. 2017. Т. 12, № 1. С. 83–87.
14. Гусев А. В., Зарубина Т. В. Поддержка принятия врачебных решений в медицинских информационных системах медицинской организации. *Врач и информационные технологии.* 2017. № 2. С. 60–72.
15. Кобринский Б. А. Проблема взаимопонимания: термины и определения в медицинской информатике. *Врач и информационные технологии*. 2009. № 1. С. 51–52.
16. Демикова Н. С., Лапина А. С., Путинцев А. Н., Шмелева Н. Н. Информационно-справочная система по врожденным порокам

развития в медицинской практике и образовании. *Врач и информационные технологии.* 2007. № 5. С. 33–36.

17. Назаренко Г. И., Осипов Г. С., Назаренко А. Г., Молодченков А. И. Интеллектуальные системы в клинической медицине. Синтез плана лечения на основе прецедентов. *Информационные технологии и вычислительные системы.* 2010. № 1. С. 24–35.
18. Polat K. Computer aided diag№sis of ECG data on the least square support vector machine. *Digit Signal Process.* 2008. Vol. 18, № 1. P. 25-32.
19. Mofidi R. Identification of severe acute pancreatitis using an artificial neural network. *Surgery.* 2007. Vol. 141, № 1 P. 59–66.
20. Джарратано Дж., Райли Г. Экспертные системы: принципы разработки и программирование. 4-е изд. Москва: Вильямс, 2007. 1152 с.
21. Джексон П. Введение в экспертные системы: учеб. пособие. Москва: Вильямс, 2001. 624 с.
22. Нейлор К. Как построить свою экспертную систему. Москва: Энергоатомиздат, 1991. 286 с.
23. Berner E. S. Clinical Decision Support Systems. New York: Springer, 2007. 278 p.
24. Малых В. Л. Системы поддержки принятия решений в медицине. *Программные системы: теория и приложения*. 2019. Т. 10, № 2 (41). С. 155–184.
25. Clinical Decision Support Systems: How They Improve Care and Cut Costs. URL: https://www.altexsoft.com/blog/clinical-decision-support-systems/ (accessed: 29.09.2020)
26. Згуровський М. З., Бідюк П. І., Терентьєв О. М., Просянкіна-Жарова Т. І. Байєсівські мережі в системах підтримки прийняття рішень: навчальний посібник. Київ: Видавниче Підприємство «Едельвейс», 2015. 300 с.

27. Бідюк П.І., Коршевнюк Л.О. Проектування комп'ютерних інформаційних систем підтримки прийняття рішень: навчальний посібник. Київ: ННК «ІПСА» НТУУ «КПІ», 2010. 340 с.
28. Heckerman D., Geiger D., Chikering D. Learning Bayesian networks: the combination of knowledge and statistical data. *Machine Learning*. 1995. No 20. P. 197–243.
29. Бідюк П.І., Коршевнюк Л.О. Методика побудови ймовірнісних мережних моделей. *Комп'ютерні технології.* 2011. Т. 160. С. 6–14.
30. HUGIN Graphical User Interface Documentation, Release 8.9, 2020. URL: http://download.hugin.com/webdocs/manuals/8.9/_downloads/545b768771d6db8e42d51ecafa712c7b/HUGIN_GUI.pdf (accessed: 15.10.2020).
31. Greedy Search-And-Score Structure Learning**.** URL: http://data.biotracer.hugin.com/htmlhelp/descr_greedy_pane.html (accessed: 15.10.2020).
32. Chow-Liu Tree. URL: http://data.biotracer.hugin.com/htmlhelp/descr_chow_liu_pane.html (accessed: 15.10.2020).
33. Tree Augmented Naive Bayes. URL: http://data.biotracer.hugin.com/htmlhelp/descr_tan_pane.html (accessed: 15.10.2020).
34. Lepar V., Shenoy P. P. A Comparison of Lauritzen-Spiegelhalter, Hugin, and Shenoy-Shafer Architectures for Computing Marginals of Probability Distributions. *Uncertainty in Artificial Intelligence*. San Francisco, CA: Morgan Kaufmann, 1999. Vol. 14. P. 328–337.
35. Agenarisk. URL: http://www.agenarisk.com/ (accessed: 18.10.2020).
36. BayesiaLab. URL: http://www.bayesia.com/ (accessed: 18.10.2020).
37. Bayes Server. URL: http://www.bayesserver.com/ (accessed: 18.10.2020).

38. Торопова А. В. Байесовские сети доверия: инструменты и использование в учебном процессе. *Компьютерные инструменты в образовании.* 2016. № 4. С. 43–53.
39. GeNIe Modeler, BayesFusion LLC. URL: http://www.bayesfusion.com/ (accessed: 18.10.2020).
40. Hugin Expert. URL: http://www.hugin.com/ (accessed: 18.10.2020).

ДОДАТОК А ЛІСТИНГ ОБРОБКИ ДАНИХ

Heart disease work with data.ipynb

```
import pandas as pd
import pandas_profiling
from pandas_profiling import ProfileReport
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
data = pd.read_csv(r"…\heart.csv")
data2 = data.copy()
plt.figure(figsize=(11,10))
sns.heatmap(data.corr(), annot = True, cbar=False, vmin=-1, vmax=1, center= 0, cmap= 'Pastel1', fmt='.2g', linewidths=1, linecolor='white', square=True)
from numpy import*
age_cat_5 = histogram(data2.age, bins=5)
trestbps_cat_5 = histogram(data2.trestbps, bins=5)
chol_cat_5 = histogram(data2.chol, bins=5)
thalach_cat_5 = histogram(data2.thalach, bins=5)
oldpeak_cat_5 = histogram(data2.oldpeak, bins=5)
ca_cat_5 = histogram(data2.ca, bins=5)
data5 = data.copy()
data5_bez_oldpeak = data5.copy()
del data5_bez_oldpeak['oldpeak_cat']
data5_bez_oldpeak.to_csv('…\Heart\data5_bez_oldpeak.csv', index=False)
category = pd.cut(data5.age,bins=age_cat_5[1],labels=['0','1','2','3', '4'], include_lowest=True)
category1 = pd.cut(data5.trestbps,bins=trestbps_cat_5[1],labels=['0','1','2','3', '4'], include_lowest=True)
category2 = pd.cut(data5.chol,bins=chol_cat_5[1],labels=['0','1','2','3', '4'], include_lowest=True)
category3 = pd.cut(data5.thalach,bins=thalach_cat_5[1],labels=['0','1','2','3', '4'], include_lowest=True)
```

```
category5 = pd.cut(data5.ca,bins=ca_cat_5[1],labels=['0','1','2','3', '4'], include_lowest=True)
data5.insert(1,'age_cat',category)
data5.insert(5,'trestbps_cat',category1)
data5.insert(7,'chol_cat',category2)
data5.insert(11,'thalach_cat',category3)
data5.insert(17,'ca_cat',category5)
del data5['age']
del data5['trestbps']
del data5['chol']
del data5['thalach']
del data5['ca']
data5.to_csv('…\Heart\data5.csv', index=False)
data5 = pd.read_csv('…\Heart\data5.csv')
data5_bez_oldpeak = data5.copy()
data5_bez_oldpeak_for_tests = data5_bez_oldpeak.copy()
del data5_bez_oldpeak_for_tests['target']
X = data5_bez_oldpeak_for_tests
y = data5_bez_oldpeak.target
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test =  train_test_split(X, y, test_size = 0.2, random_state= 0)
data5_bez_oldpeak_train = X_train
data5_bez_oldpeak_train.insert(12,'target',y_train)
data5_bez_oldpeak_train.to_csv('…\Heart\data5_bez_oldpeak_train.csv', index=False)
data5_bez_oldpeak_test = X_test
data5_bez_oldpeak_test.insert(12,'target',y_test)
data5_bez_oldpeak_test.to_csv('…\Heart\data5_bez_oldpeak_test.csv', index=False)
```

Covid work with data.ipynb

```
import pandas as pd
import pandas_profiling
from pandas_profiling import ProfileReport
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
```

```
data = pd.read_csv(r'…\Covid\Covid Dataset.csv')
data.profile_report()
data.drop_duplicates(keep='first',inplace=True)
del data['Wearing Masks']
del data['Sanitization from Market']
data.to_csv(r'…\Covid\data_bez_odnac.csv', index=False)
data_bez_odnac = pd.read_csv(r'…\Covid\data_bez_odnac.csv')
data_bez_odnac['Fatigue '] = data_bez_odnac['Fatigue '].map({'Yes': 1, 'No': 0})
data_bez_odnac.to_csv(r'…\Covid\data_10.csv', index=False)
data = pd.read_csv(r'…\Covid\data_10.csv')
data_short = data.copy()
del data_short['Asthma']
del data_short['Chronic_Lung_Disease']
del data_short['Heart_Disease']
del data_short['Diabetes']
del data_short['Hyper_Tension']
data_short.to_csv(r'…\Covid\data_short.csv', index=False)
data_short = pd.read_csv(r'…\Covid\data_short.csv')
data_without_target = data_short.copy()
del data_without_target['target']
X = data_without_target
Y = data_short['target']
X_train, X_test, y_train, y_test =  train_test_split(X,Y,test_size = 0.2, random_state= 0)
data_train = X_train
data_train.insert(12,'target',y_train)
data_train.to_csv(r'…\Covid\data_train.csv', index=False)
data_test = X_test
data_test.insert(12,'target',y_test)
data_test.to_csv(r'…\Covid\data_test.csv', index=False)
```

ДОДАТОК Б ТАБЛИЦІ ЙМОВІРНОСТЕЙ ДЛЯ СИСТЕМИ ДІАГНОСТИКИ ХВОРОБ СЕРЦЯ

Таблиця Б.1 – Таблиця безумовних ймовірностей для вузла age_cat

0	0.041322
1	0.214876
2	0.31405
3	0.376033
4	0.053719
Experience	242

Таблиця Б.2 – Таблиця безумовних ймовірностей для вузла sex

0	0.326446
1	0.673554
Experience	242

Таблиця Б.3 – Таблиця умовних ймовірностей для вузла target

sex	0					1				
age_cat	0	1	2	3	4	0	1	2	3	4
0	0	0.071429	0.181818	0.382353	0	0.285714	0.368421	0.555556	0.736842	0.714286
1	1	0.928571	0.818182	0.617647	1	0.714286	0.631579	0.444444	0.263158	0.285714
Experience	3	14	22	34	6	7	38	54	57	7

Таблиця Б.4 – Таблиця умовних ймовірностей для вузла ca_cat

sex	0									
age_cat	0		1		2		3		4	
target	0	1	0	1	0	1	0	1	0	1
0	0.2	1	1	0.923077	0.75	0.833333	0.307692	0.666667	0.2	0.5
1	0.2	0	0	0.076923	0.25	0.166667	0	0.190476	0.2	0.166667
2	0.2	0	0	0	0	0	0.384615	0.142857	0.2	0.333333
3	0.2	0	0	0	0	0	0.307692	0	0.2	0
4	0.2	0	0	0	0	0	0	0	0.2	0
Experience	0	3	1	13	4	18	13	21	0	6

Кінець таблиці Б.4

sex	1									
age_cat	0		1		2		3		4	
target	0	1	0	1	0	1	0	1	0	1
0	1	0.6	0.642857	0.916667	0.266667	0.875	0.190476	0.866667	0.2	0
1	0	0	0.142857	0.041667	0.533333	0.083333	0.333333	0.066667	0	1
2	0	0	0.142857	0.041667	0.066667	0	0.380952	0	0.2	0
3	0	0	0	0	0.133333	0.041667	0.095238	0	0.6	0
4	0	0.4	0.071429	0	0	0	0	0.066667	0	0
Experience	2	5	14	24	30	24	42	15	5	2

Таблиця Б.5 – Таблиця умовних ймовірностей для вузла thal

sex	0		1	
target	0	1	0	1
0	0	0.016393	0.010753	0
1	0.055556	0	0.11828	0.057143
2	0.388889	0.95082	0.236559	0.671429
3	0.555556	0.032787	0.634409	0.271429
Experience	18	61	93	70

Таблиця Б.6 – Таблиця умовних ймовірностей для вузла cp

target	0	1
0	0.756757	0.251908
1	0.063063	0.221374
2	0.126126	0.419847
3	0.054054	0.10687
Experience	111	131

Таблиця Б.7 – Таблиця умовних ймовірностей для вузла trestbps_cat

age_cat	0		1		2		3		4	
target	0	1	0	1	0	1	0	1	0	1
0	0	0	0.4	0.324324	0.088235	0.166667	0.145455	0.138889	0	0.25
1	1	0.625	0.4	0.432432	0.5	0.595238	0.363636	0.333333	0.4	0.25
2	0	0.375	0.2	0.243243	0.294118	0.214286	0.345455	0.416667	0.4	0.25
3	0	0	0	0	0.058824	0.02381	0.145455	0.083333	0	0.25
4	0	0	0	0	0.058824	0	0	0.027778	0.2	0
Experience	2	8	15	37	34	42	55	36	5	8

Таблиця Б.8 – Таблиця умовних ймовірностей для вузла fbs

target	0	1
0	0.828829	0.870229
1	0.171171	0.129771
Experience	111	131

Таблиця Б.9 – Таблиця умовних ймовірностей для вузла restecg

target	0	1
0	0.54955	0.450382
1	0.423423	0.541985
2	0.027027	0.007634
Experience	111	131

Таблиця Б.10 – Таблиця умовних ймовірностей для вузла slope

target	0	1
0	0.081081	0.061069
1	0.630631	0.29771
2	0.288288	0.641221
Experience	111	131

Таблиця Б.11 – Таблиця умовних ймовірностей для вузла exang

cp	0		1		2		3	
target	0	1	0	1	0	1	0	1
0	0.321429	0.69697	0.857143	0.931034	0.785714	0.909091	0.833333	0.857143
1	0.678571	0.30303	0.142857	0.068965	0.214286	0.090909	0.166667	0.142857
Experience	84	33	7	29	14	55	6	14

Таблиця Б.12 – Таблиця умовних ймовірностей для вузла chol_cat

sex	0											
age_cat	0						1					
target	0			1			0			1		
restecg	0	1	2	0	1	2	0	1	2	0	1	2
0	0.2	0.2	0.2	0.2	0.7	0.2	0	0.2	0.2	0.3	0.857143	0.2
1	0.2	0.2	0.2	0.2	0.3	0.2	0	0.2	0.2	0.7	0.142857	0.2
2	0.2	0.2	0.2	0.2	0	0.2	1	0.2	0.2	0	0	0.2
3	0.2	0.2	0.2	0.2	0	0.2	0	0.2	0.2	0	0	0.2
4	0.2	0.2	0.2	0.2	0	0.2	0	0.2	0.2	0	0	0.2
Experience	0	0	0	0	3	0	1	0	0	6	7	0

Продовження таблиці Б.12

sex	0											
age_cat	2						3					
target	0			1			0			1		
restecg	0	1	2	0	1	2	0	1	2	0	1	2
0	0.2	0	0.5	0.083333	0.166667	0.2	0.125	0.2	0.2	0	0.285714	0.2
1	0.2	0.5	0	0.833333	0.333333	0.2	0.375	0.8	0.2	0.428571	0.357143	0.2
2	0.2	0.5	0.5	0.083333	0.5	0.2	0.375	0	0.2	0.142857	0.357143	0.2
3	0.2	0	0	0	0	0.2	0.125	0	0.2	0.285714	0	0.2
4	0.2	0	0	0	0	0.2	0	0	0.2	0.142857	0	0.2
Experience	0	2	2	12	6	0	8	5	0	7	14	0

Продовження таблиці Б.12

sex	0						1					
age_cat	4						0					
target	0			1			0			1		
restecg	0	1	2	0	1	2	0	1	2	0	1	2
0	0.2	0.2	0.2	0.5	0.333333	1	0	0	0.2	1	0.666667	0.2
1	0.2	0.2	0.2	0.5	0.333333	0	1	1	0.2	0	0.333333	0.2
2	0.2	0.2	0.2	0	0.333333	0	0	0	0.2	0	0	0.2
3	0.2	0.2	0.2	0	0	0	0	0	0.2	0	0	0.2
4	0.2	0.2	0.2	0	0	0	0	0	0.2	0	0	0.2
Experience	0	0	0	2	3	1	1	1	0	2	3	0

Продовження таблиці Б.12

sex	1											
age_cat	1						2					
target	0			1			0			1		
restecg	0	1	2	0	1	2	0	1	2	0	1	2
0	0.5	0.125	0.2	0.111111	0.4	0.2	0.266667	0.266667	0.2	0.272727	0.461538	0.2
1	0.5	0.625	0.2	0.777778	0.466667	0.2	0.733333	0.6	0.2	0.727273	0.461538	0.2
2	0	0.25	0.2	0.111111	0.133333	0.2	0	0.13333	0.2	0	0.076923	0.2
3	0	0	0.2	0	0	0.2	0	0	0.2	0	0	0.2
4	0	0	0.2	0	0	0.2	0	0	0.2	0	0	0.2
Experience	6	8	0	9	15	0	15	15	0	11	13	0

Кінець таблиці Б.12

sex	1											
age_cat	3						4					
target	0			1			0			1		
restecg	0	1	2	0	1	2	0	1	2	0	1	2
0	0.307692	0.2	0	0.333333	0.166667	0.2	0	1	0.2	0	0	0.2
1	0.653846	0.666667	0	0.555556	0.833333	0.2	0.5	0	0.2	1	1	0.2
2	0.038462	0.133333	1	0.111111	0	0.2	0.5	0	0.2	0	0	0.2
3	0	0	0	0	0	0.2	0	0	0.2	0	0	0.2
4	0	0	0	0	0	0.2	0	0	0.2	0	0	0.2
Experience	26	15	1	9	6	0	4	1	0	1	1	0

Таблиця Б.13 – Таблиця умовних ймовірностей для вузла thalach_cat

exang	0									
slope	0									
age_cat	0		1		2		3		4	
target	0	1	0	1	0	1	0	1	0	1
0	0.2	0	0	0.2	0.2	0	0	0	0.2	0.2
1	0.2	0	0	0.2	0.2	0	0	0.333333	0.2	0.2
2	0.2	0	0	0.2	0.2	0	1	0.333333	0.2	0.2
3	0.2	0	1	0.2	0.2	1	0	0.333333	0.2	0.2
4	0.2	1	0	0.2	0.2	0	0	0	0.2	0.2
Experience	0	1	1	0	0	3	2	3	0	0

Продовження таблиці Б.13

exang	0									
slope	1									
age_cat	0		1		2		3		4	
target	0	1	0	1	0	1	0	1	0	1
0	0.2	0.2	0	0	0	0	0.125	0	0	0
1	0.2	0.2	0	0.125	0.2	0	0.1875	0.125	0.333333	0.5
2	0.2	0.2	1	0.125	0.6	0.181818	0.1875	0.25	0.666667	0.5
3	0.2	0.2	0	0.5	0.2	0.636364	0.5	0.625	0	0
4	0.2	0.2	0	0.25	0	0.181818	0	0	0	0
Experience	0	0	2	8	5	11	16	8	3	4

Продовження таблиці Б.13

exang	0									
slope	2									
age_cat	0		1		2		3		4	
target	0	1	0	1	0	1	0	1	0	1
0	0.2	0	0	0	0	0	0	0.052632	0.2	0
1	0.2	0	0	0	0	0.095238	0	0	0.2	0
2	0.2	0	0	0.086957	0.4	0.095238	0.1	0.315789	0.2	0.25
3	0.2	0.571429	0.6	0.521739	0.4	0.714286	0.9	0.526316	0.2	0.75
4	0.2	0.428571	0.4	0.391304	0.2	0.095238	0	0.105263	0.2	0
Experience	0	7	5	23	5	21	10	19	0	4

Продовження таблиці Б.13

exang	1									
slope	0									
age_cat	0		1		2		3		4	
target	0	1	0	1	0	1	0	1	0	1
0	0.2	0.2	0	0	0	0.2	0.5	0.2	0.2	0.2
1	0.2	0.2	0	0	0.666667	0.2	0	0.2	0.2	0.2
2	0.2	0.2	1	0	0	0.2	0.5	0.2	0.2	0.2
3	0.2	0.2	0	1	0.333333	0.2	0	0.2	0.2	0.2
4	0.2	0.2	0	0	0	0.2	0	0.2	0.2	0.2
Experience	0	0	1	1	3	0	2	0	0	0

Продовження таблиці Б.13

exang	1									
slope	1									
age_cat	0		1		2		3		4	
target	0	1	0	1	0	1	0	1	0	1
0	0	0.2	0	0	0.105263	0	0	0	0	0.2
1	0	0.2	0.4	0	0.421053	0	0.111111	0.25	0	0.2
2	0	0.2	0.6	0.333333	0.421053	1	0.722222	0.5	0	0.2
3	0	0.2	0	0.666667	0.052632	0	0.166667	0.25	1	0.2
4	1	0.2	0	0	0	0	0	0	0	0.2
Experience	1	0	5	3	19	1	18	4	1	0

Продовження таблиці Б.13

exang	1									
slope	2									
age_cat	0		1		2		3		4	
target	0	1	0	1	0	1	0	1	0	1
0	0	0.2	0	0	0	0	0	0	0	0.2
1	0	0.2	0	0	0	0.16667	0.14286	0	0	0.2

Кінець таблиці Б.13

exang	1									
slope	2									
age_cat	0		1		2		3		4	
target	0	1	0	1	0	1	0	1	0	1
2	0	0.2	0	0	0	0	0.571429	0	0	0.2
3	1	0.2	1	0	1	0.666667	0.285714	1	1	0.2
4	0	0.2	0	1	0	0.166667	0	0	0	0.2
Experience	1	0	1	2	2	6	7	2	1	0

ДОДАТОК В ТАБЛИЦІ ЙМОВІРНОСТЕЙ ДЛЯ СИСТЕМИ ДІАГНОСТИКИ COVID-19

Таблиця В.1 – Таблиця безумовних ймовірностей для вузла Attended_Large_Gathering

0	0.564516
1	0.435484
Experience	372

Таблиця В.2 – Таблиця безумовних ймовірностей для вузла Abroad_travel

0	0.575269
1	0.424731
Experience	372

Таблиця В.3 – Таблиця безумовних ймовірностей для вузла Family_working_in_Public_Exposed_Places

0	0.553763
1	0.446237
Experience	372

Таблиця В.4 – Таблиця безумовних ймовірностей для вузла Visited_Public_Exposed_Places

0	0.52957
1	0.47043
Experience	372

Таблиця В.5 – Таблиця умовних ймовірностей для вузла Contact_with_COVID_Patient

Attended_Larg	0	1
0	0.57619	0.462963
1	0.42381	0.537037

Таблиця В.6 – Таблиця умовних ймовірностей для вузла target

Visited_Public_	0									
Family_workin	0								1	
Attended_Larg	0				1				0	
Abroad_travel	0		1		0		1		0	
Contact_with_	0	1	0	1	0	1	0	1	0	1
0	0.571429	0.35	0	0	0.090909	0.2	0	0	0.4	0.071429
1	0.428571	0.65	1	1	0.909091	0.8	1	1	0.6	0.928571
Experience	28	20	18	5	11	15	4	6	15	14

Продовження таблиці В.6

Visited_Public_	0						1			
Family_workin	1						0			
Attended_Larg	0		1				0			
Abroad_travel	1		0		1		0		1	
Contact_with_	0	1	0	1	0	1	0	1	0	1
0	0	0	0.222222	0.285714	0	0	0.695652	0	0	0
1	1	1	0.777778	0.714286	1	1	0.304348	1	1	1

Продовження таблиці В.6

Visited_Public_	1									
Family_workin	0				1					
Attended_Larg	1				0				1	
Abroad_travel	0		1		0		1		0	
Contact_with_	0	1	0	1	0	1	0	1	0	1
0	0.111	0.058824	0	0	0.571429	0	0	0	0.285714	0.111
1	0.889	0.941176	1	1	0.428571	1	1	1	0.714286	0.889
Experience	9	17	10	10	14	9	8	7	7	9

Кінець таблиці В.6

Visited_Public_	1	
Family_workin	1	
Attended_Larg	1	
Abroad_travel	1	
Contact_with_	0	1
0	0	0
1	1	1
Experience	12	10

Таблиця В.7 – Таблиця умовних ймовірностей для вузла Headache

target	0	1
0	0.492537	0.504918
1	0.507463	0.495082
Experience	67	305

Таблиця В.8 – Таблиця умовних ймовірностей для вузла Fatigue

target	0	1
0	0.38806	0.504918
1	0.61194	0.495082
Experience	67	305

Таблиця В.9 – Таблиця умовних ймовірностей для вузла Sore_throat

target	0	1
0	0.746269	0.301639
1	0.253731	0.698361
Experience	67	305

Таблиця В.10 – Таблиця умовних ймовірностей для вузла Gastrointestinal

target	0	1
0	0.522388	0.531148
1	0.477612	0.468852

Таблиця В.11 – Таблиця умовних ймовірностей для вузла Dry_Cough

target	0	1
0	0.701493	0.206557
1	0.298507	0.793443
Experience	67	305

Таблиця В.12 – Таблиця умовних ймовірностей для вузла Running_Nose

target	0	1
0	0.358209	0.52459
1	0.641791	0.47541
Experience	67	305

Таблиця В.13 – Таблиця умовних ймовірностей для вузла Fever

Sore_throat	0		1	
target	0	1	0	1
0	0.62	0.271739	0.294118	0.211268
1	0.38	0.728261	0.705882	0.788732
Experience	50	92	17	213

Таблиця В.14 – Таблиця умовних ймовірностей для вузла Breathing_Problem

Sore_throat	0								1	
target	0				1				0	
Dry_Cough	0		1		0		1		0	
Running_Nose	0	1	0	1	0	1	0	1	0	1
0	0.833333	0.96	1	0.5	0	0.25	0.418605	0.22727 3	0.666667	0.714286
1	0.166667	0.04	0	0.5	1	0.75	0.581395	0.77272 7	0.333333	0.285714
Experience	12	25	5	8	1	4	43	44	3	7

Кінець таблиці В.14

Sore_throat	1					
target	0		1			
Dry_Cough	1		0		1	
Running_Nose	0	1	0	1	0	1
0	0.5	0.333333	0	0.043478	0.308642	0.378378
1	0.5	0.666667	1	0.956522	0.691358	0.621622
Experience	4	3	35	23	81	74

ДОДАТОК Г НАУКОВІ ПУБЛІКАЦІЇ

1. Корнійчук О. С. Медична діагностична система на основі байєсівських мереж. Проблеми інформатизації: матеріали восьмої міжнародної науково-технічної конференції (Харків, 26 листопада – 27 листопада 2020 р.). Харків, 2020. Т. 3. С. 17.

2. Корнійчук О. С., Бідюк П. І. Медична діагностична система на основі байєсівських мереж. *Системні науки і кібернетика*. 2020. С. 82–105.